XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: FpML
- Global Definitions
  - Complex Type: ValuationDocument
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ○ fpml-fx-4-8.xsd
	  ○ fpml-ird-4-8.xsd
	  ○ fpml-eqd-4-8.xsd
	  ○ fpml-return-swaps-4-8.xsd
	  ○ fpml-cd-4-8.xsd
	  ○ fpml-bond-option-4-8.xsd
	  ○ fpml-correlation-swaps-4-8.xsd
	  ○ fpml-dividend-swaps-4-8.xsd
	  ○ fpml-variance-swaps-4-8.xsd
	  ○ fpml-loan-4-8.xsd
	  ○ fpml-com-4-8.xsd
	  ○ fpml-pretrade-4-8.xsd
	  ○ fpml-tradeexec-4-8.xsd
	  ○ fpml-posttrade-negotiation-4-8.xsd
	  ○ fpml-posttrade-execution-4-8.xsd
	  ○ fpml-allocation-4-8.xsd
	  ○ fpml-trade-notification-4-8.xsd
	  ○ fpml-contract-notification-4-8.xsd
	  ○ fpml-confirmation-4-8.xsd
	  ○ fpml-posttrade-confirmation-4-8.xsd
	  ○ fpml-credit-event-notification-4-8.xsd
	  ○ fpml-reporting-4-8.xsd
	  ○ fpml-reconciliation-4-8.xsd
	  ○ fpml-matching-status-4-8.xsd

Documentation	products business process messaging pre-trade

	negotiation and execution notification confirmation reporting and settlement miscellaneous

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-fx-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-ird-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-eqd-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-return-swaps-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-cd-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-bond-option-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-correlation-swaps-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-dividend-swaps-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-variance-swaps-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-loan-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-com-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-pretrade-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-tradeexec-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-posttrade-negotiation-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-posttrade-execution-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-allocation-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-trade-notification-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-contract-notification-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-confirmation-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-posttrade-confirmation-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-credit-event-notification-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-reporting-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-reconciliation-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-matching-status-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: Schema Component Type	AusAddress Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope.

See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: FpML

[Table of contents]

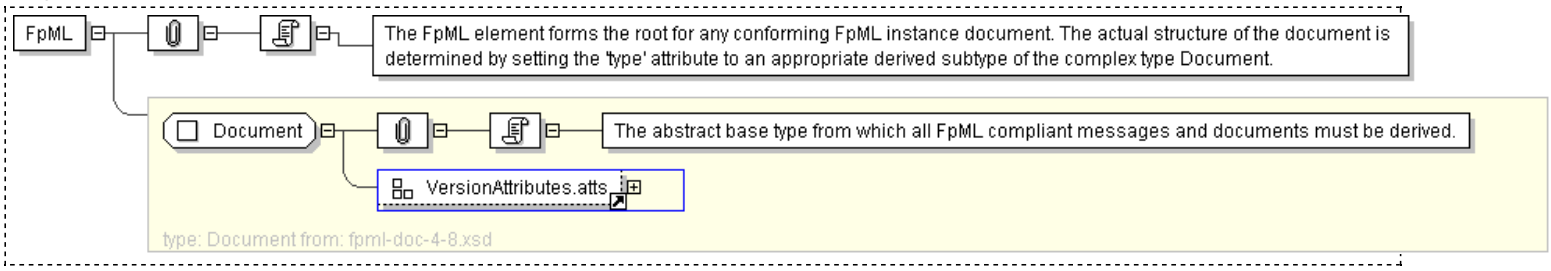
Name	FpML
Type	Document
Nillable	no
Abstract	no
Documentation	The FpML element forms the root for any conforming FpML instance document. The actual structure of the document is determined by setting the 'type' attribute to an appropriate derived subtype of the complex type Document.

Logical Diagram

XML Instance Representation

```
<FpML
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
the actual build number stays the same.'

 "/>
```

Diagram



Schema Component Representation

```
<xsd:element name="FpML" type=" Document "/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationDocument

[Table of contents]

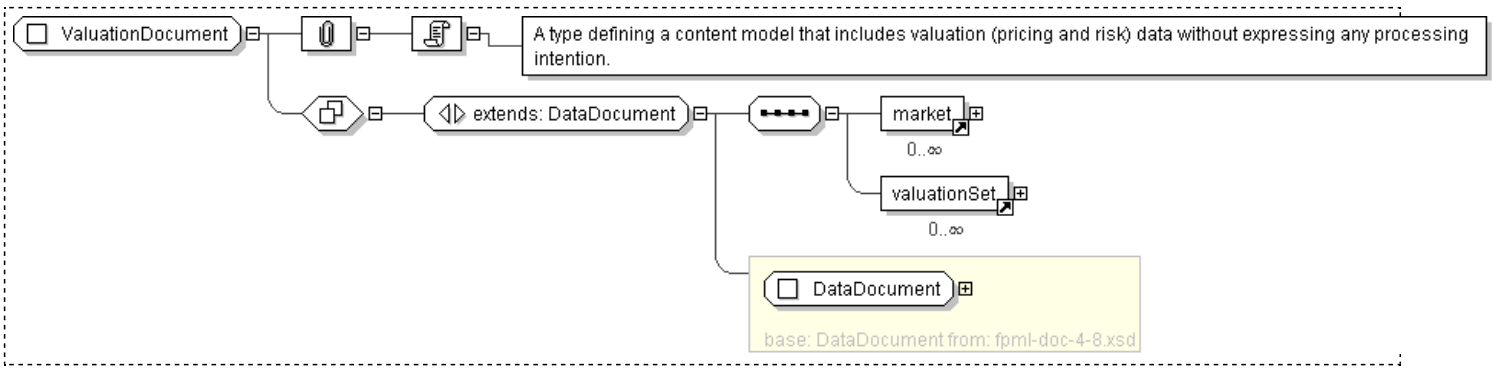
Super-types:	DataDocument < ValuationDocument (by extension)
Sub-types:	None

Name	ValuationDocument
Abstract	no
Documentation	A type defining a content model that includes valuation (pricing and risk) data without expressing any processing intention.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
 actual build number stays the same.'

 ">
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [0..*]
       'The root element in an FpML trade document.'

      <portfolio> Portfolio </portfolio> [0..*]
       'An arbitary grouping of trade references (and possibly other portfolios).'

      <event> ... </event> [1..*]
       'A business event.'

   End Choice
   <party> Party </party> [0..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term
   of the trade, but may include other parties involved in, or incidental to, the trade, such as parties
   acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are
   defined in multiple places within a document.'

   <market> ... </market> [0..*]
   <valuationSet> ... </valuationSet> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationDocument">
  <xsd:complexContent>
    <xsd:extension base=" DataDocument ">
      <xsd:sequence>
        <xsd:element ref=" market " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref=" valuationSet " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: AllocationAmended
  - Complex Type: AllocationCancelled
  - Complex Type: AllocationCreated
  - Complex Type: RequestAllocation
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target

namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AllocationAmended

[Table of contents]

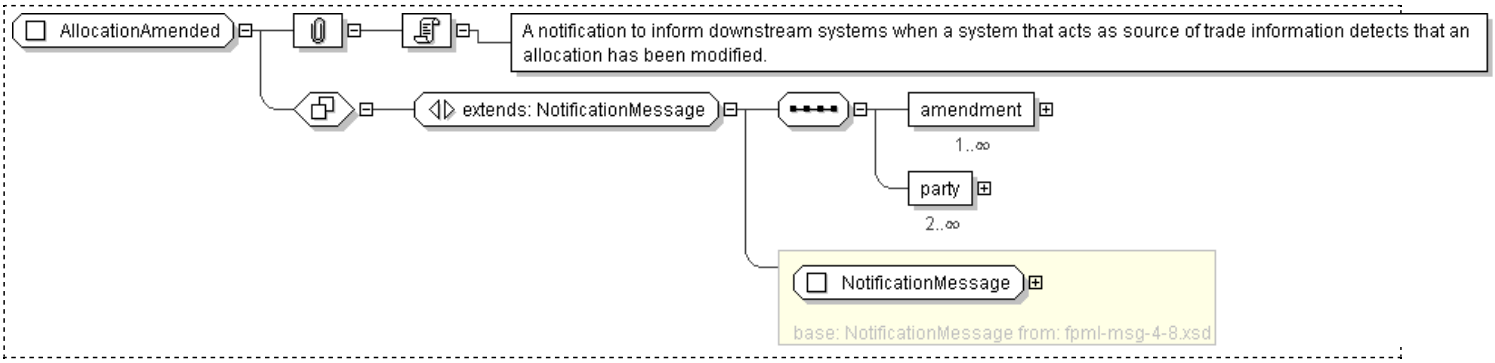
Super-types:	NotificationMessage < AllocationAmended (by extension)
Sub-types:	None

Name	AllocationAmended
Abstract	no
Documentation	A notification to inform downstream systems when a system that acts as source of trade information detects that an allocation has been modified.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> TradeAmendment </amendment> [1..*]
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AllocationAmended">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="amendment" type=" TradeAmendment " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AllocationCancelled

[Table of contents]

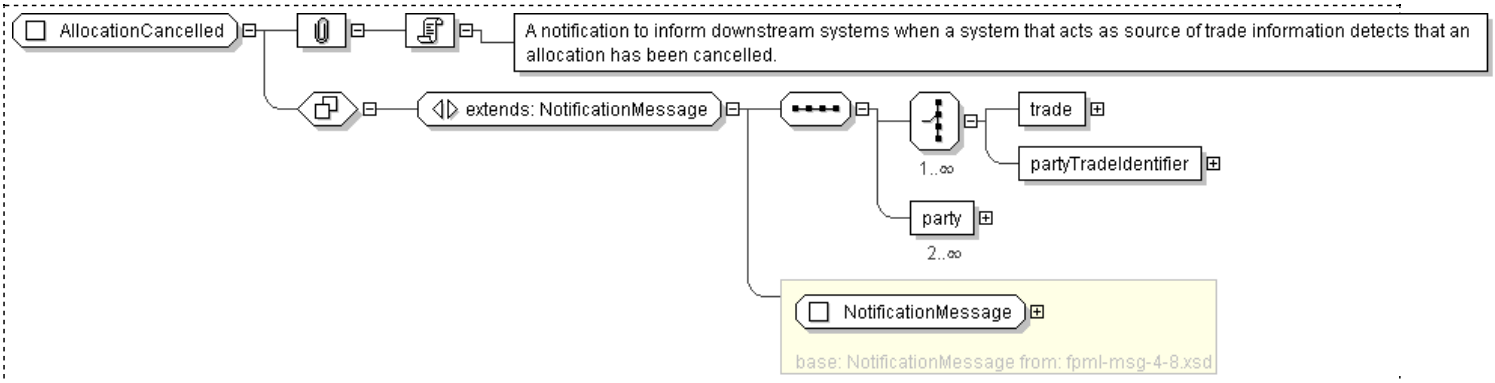
Super-types:	NotificationMessage < AllocationCancelled (by extension)
Sub-types:	None

Name	AllocationCancelled
Abstract	no
Documentation	A notification to inform downstream systems when a system that acts as source of trade information detects that an allocation has been cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1..*]
      <trade> Trade </trade> [1]
      <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1]
   End Choice
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AllocationCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:choice maxOccurs="unbounded">
          <xsd:element name="trade" type=" Trade "/>
          <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier "/>
        </xsd:choice>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AllocationCreated

[Table of contents]

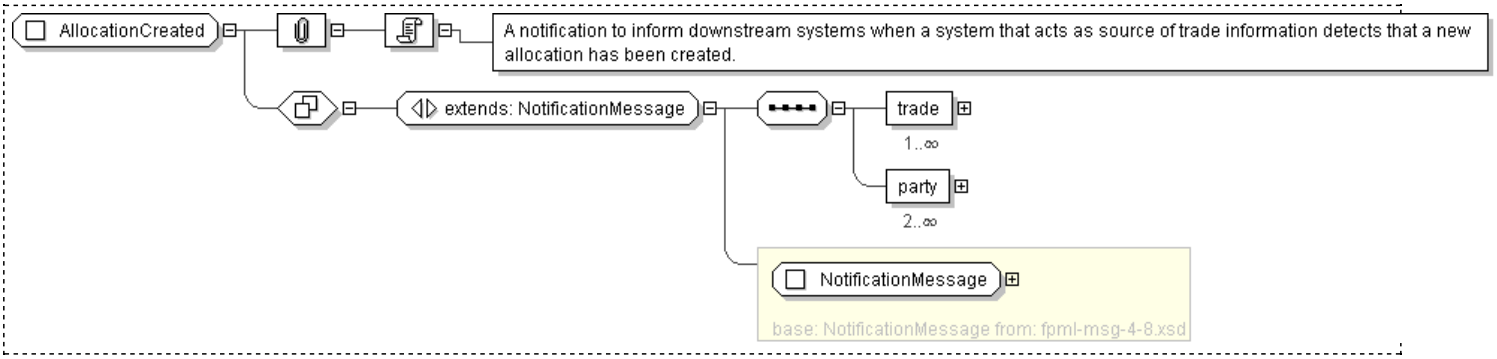
Super-types:	NotificationMessage < AllocationCreated (by extension)
Sub-types:	None

Name	AllocationCreated
Abstract	no
Documentation	A notification to inform downstream systems when a system that acts as source of trade information detects that a new allocation has been created.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1..*]
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AllocationCreated">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestAllocation

[Table of contents]

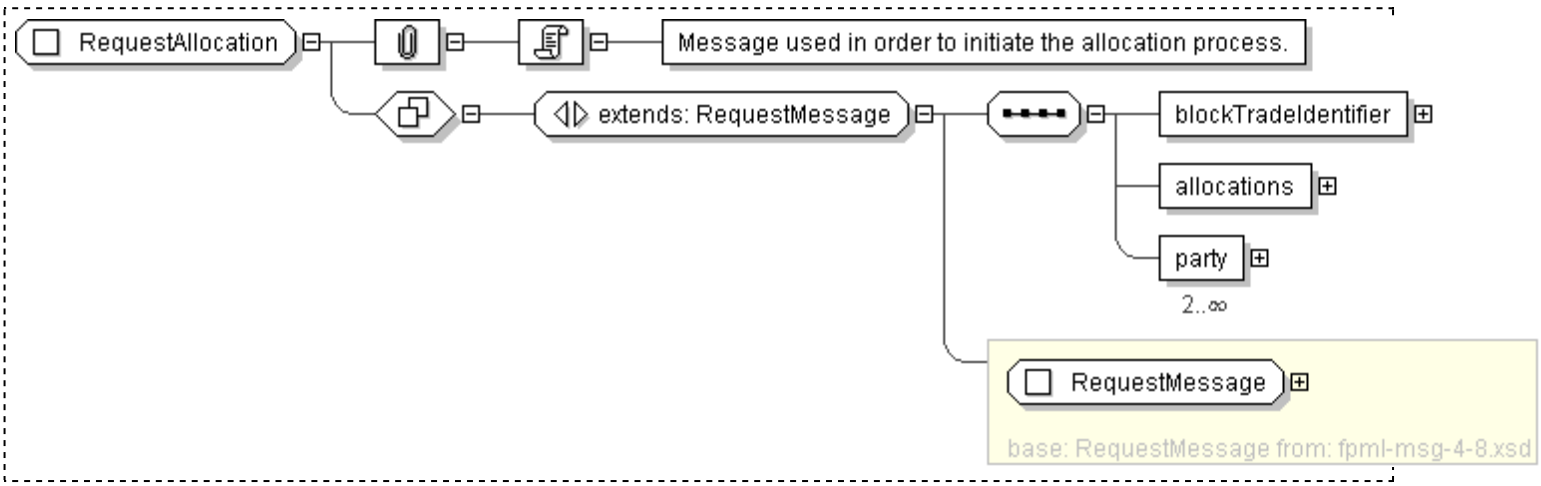
Super-types:	RequestMessage < RequestAllocation (by extension)
Sub-types:	None

Name	RequestAllocation
Abstract	no
Documentation	Message used in order to initiate the allocation process.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <blockTradeIdentifier> BlockTradeIdentifier </blockTradeIdentifier> [1]
   <allocations> Allocations </allocations> [1]
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestAllocation">
   <xsd:complexContent>
      <xsd:extension base=" RequestMessage ">
         <xsd:sequence>
            <xsd:element name="blockTradeIdentifier" type=" BlockTradeIdentifier "/>
            <xsd:element name="allocations" type=" Allocations "/>
            <xsd:element name="party" type=" Party " minOccurs="2"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: basket
  - Element: bond
  - Element: cash
  - Element: commodity
  - Element: convertibleBond
  - Element: deposit
  - Element: equity
  - Element: exchangeTradedFund
  - Element: future
  - Element: fxRate
  - Element: index
  - Element: loan
  - Element: mortgage
  - Element: mutualFund
  - Element: rateIndex
  - Element: simpleCreditDefaultSwap
  - Element: simpleFra
  - Element: simpleIrSwap
  - Element: underlyingAsset
- Global Definitions
  - Complex Type: ActualPrice
  - Complex Type: AnyAssetReference
  - Complex Type: Asset
  - Complex Type: AssetMeasureType
  - Complex Type: AssetPool
  - Complex Type: AssetReference
  - Complex Type: BasicQuotation
  - Complex Type: Basket
  - Complex Type: BasketConstituent
  - Complex Type: BasketId
  - Complex Type: BasketName
  - Complex Type: Bond
  - Complex Type: Cash
  - Complex Type: Commission
  - Complex Type: Commodity
  - Complex Type: CommodityBase
  - Complex Type: CommodityBusinessCalendar
  - Complex Type: CommodityBusinessCalendarTime
  - Complex Type: CommodityDetails
  - Complex Type: ConstituentWeight
  - Complex Type: ConvertibleBond
  - Complex Type: CouponType
  - Complex Type: Deposit
  - Complex Type: DividendPayout
  - Complex Type: EquityAsset
  - Complex Type: ExchangeTraded
  - Complex Type: ExchangeTradedCalculatedPrice
  - Complex Type: ExchangeTradedContract
  - Complex Type: ExchangeTradedFund
  - Complex Type: FacilityType
  - Complex Type: Future
  - Complex Type: FutureId

- Complex Type: FxConversion
- Complex Type: FxRateAsset
- Complex Type: IdentifiedAsset
- Complex Type: Index
- Complex Type: Lien
- Complex Type: Loan
- Complex Type: Mortgage
- Complex Type: MortgageSector
- Complex Type: MutualFund
- Complex Type: PendingPayment
- Complex Type: Price
- Complex Type: PriceQuoteUnits
- Complex Type: QuantityUnit
- Complex Type: QuotationCharacteristics
- Complex Type: QuoteTiming
- Complex Type: RateIndex
- Complex Type: SimpleCreditDefaultSwap
- Complex Type: SimpleFra
- Complex Type: SimpleIRSwap
- Complex Type: SingleUnderlyer
- Complex Type: TimeZone
- Complex Type: Underlyer
- Complex Type: UnderlyingAsset
- Complex Type: UnderlyingAssetTranche
- Model Group: BasketIdentifier.model
- Model Group: BondCalculation.model
- Model Group: BondChoice.model
- Model Group: BondContent.model
- Model Group: CommodityProduct.model
- Model Group: CommodityReferencePriceFramework.model
- Model Group: CreditEntity.model
- Model Group: EquityPrice.model
- Model Group: ExchangeIdentifier.model
- Model Group: Quotation.model
- Model Group: QuotationCharacteristics.model
- Model Group: QuoteLocation.model

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress
Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.

- If the element/attribute's type is in the schema, a link is provided to it.
- For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type,

NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

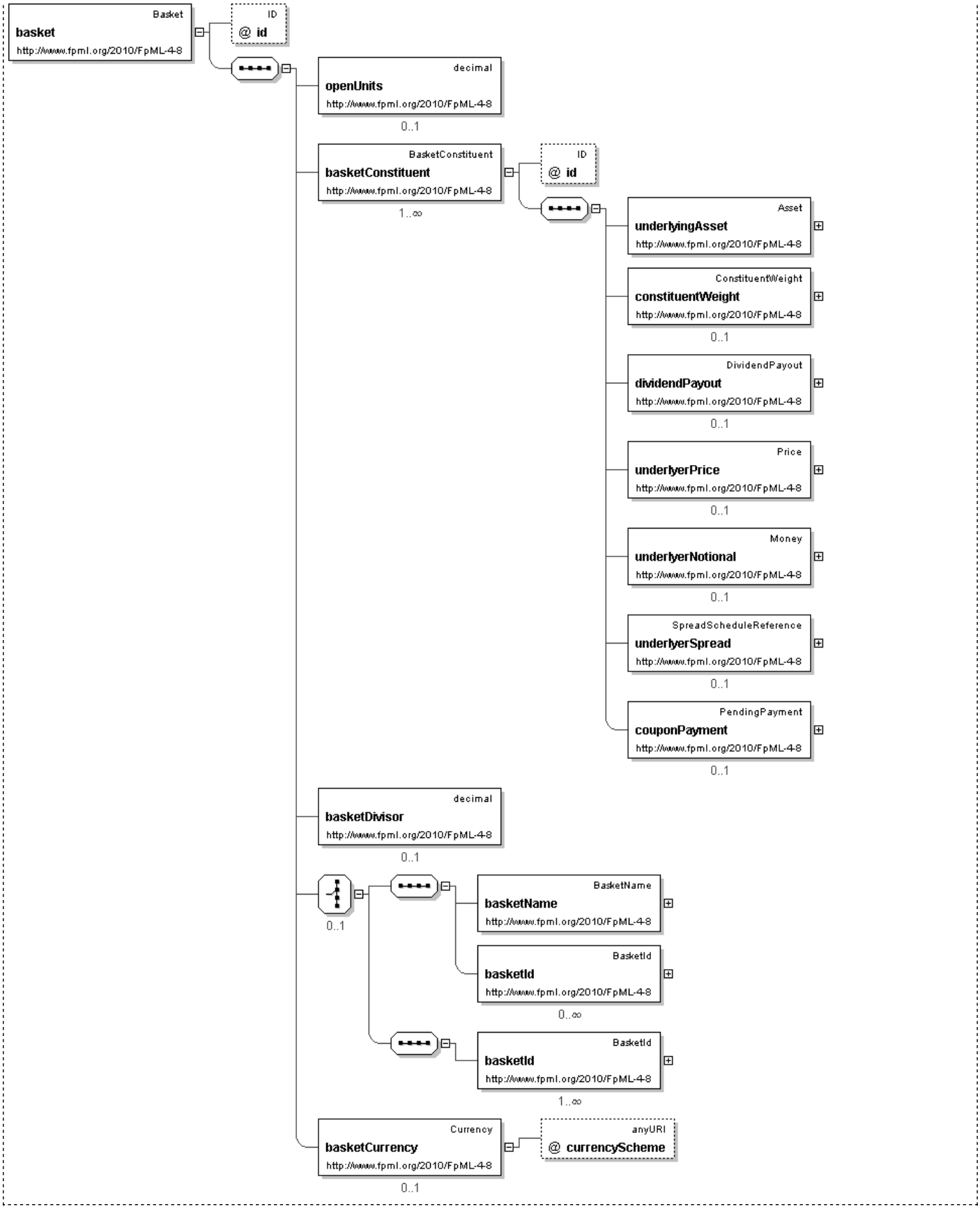
Element: basket

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	basket
Type	Basket
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a basket.

Logical Diagram

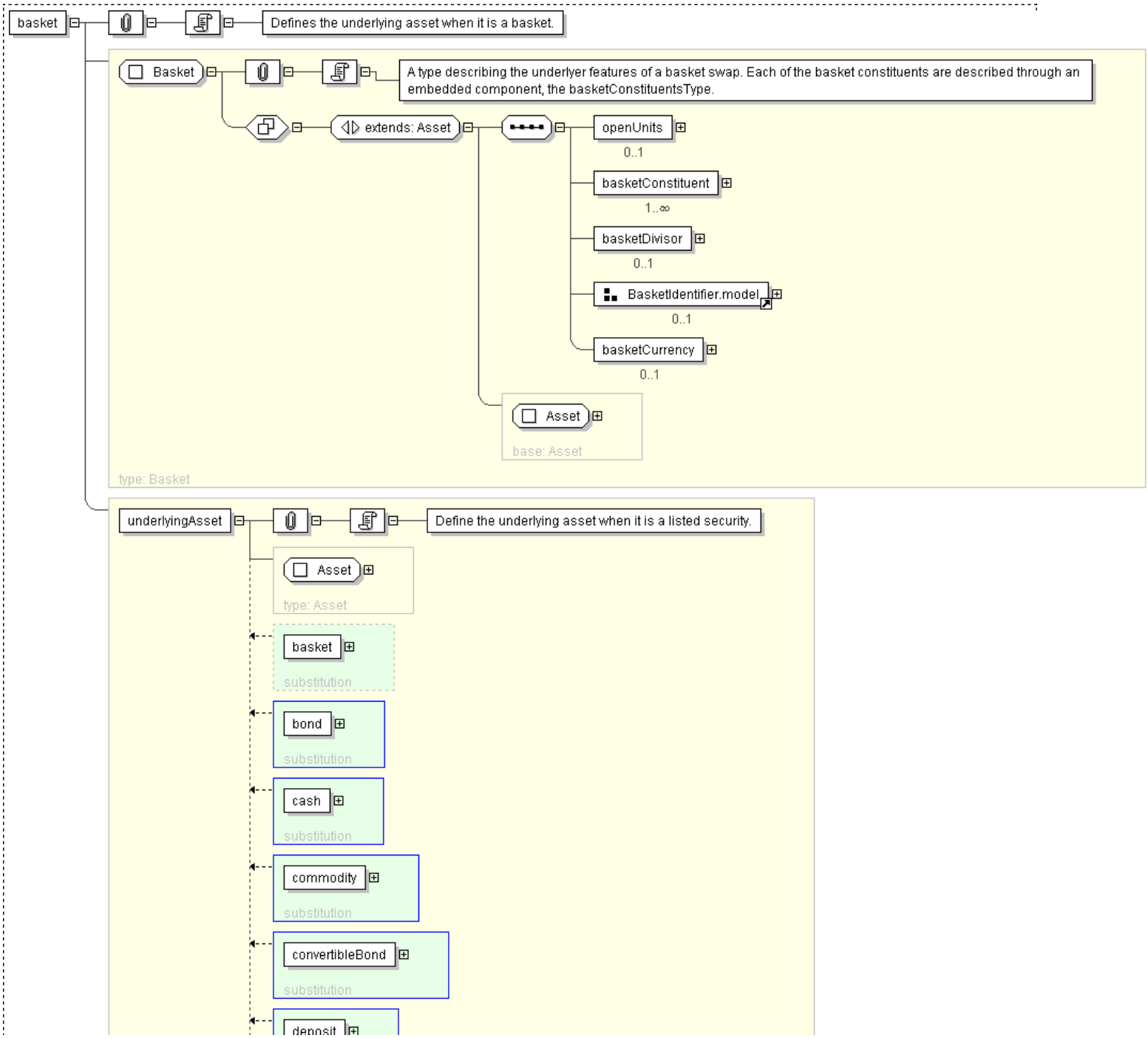


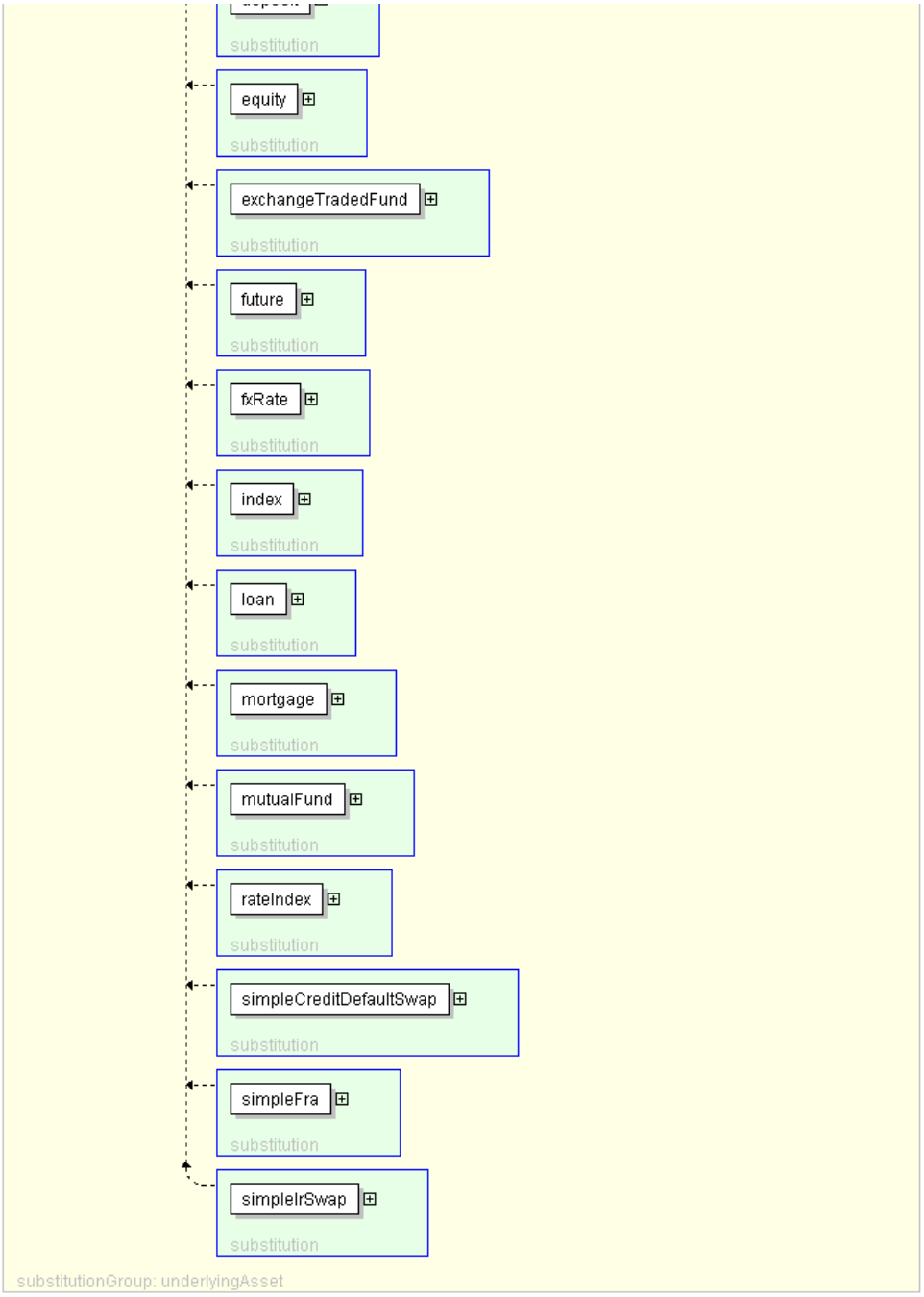
XML Instance Representation

```
<basket
 id=" xsd:ID [0..1]">
   <openUnits> xsd:decimal </openUnits> [0..1]
   'The number of units (index or securities) that constitute the underlyer of the swap. In the case of a basket
   swap, this element is used to reference both the number of basket units, and the number of each asset components
   of the basket when these are expressed in absolute terms.'

   <basketConstituent> BasketConstituent </basketConstituent> [1..*]
   'Describes each of the components of the basket.'
```

```
<basketDivisor> xsd:decimal </basketDivisor> [0..1]
'Specifies the basket divisor amount. This value is normally used to adjust the constituent weight for pricing or
to adjust for dividends, or other corporate actions.'

Start Group: BasketIdentifier.model [0..1]
'Reuses the group that specifies a name and an identifier for a given basket.'


   Start Choice [1]
      <basketName> BasketName </basketName> [1]
      'The name of the basket expressed as a free format string. FpML does not define usage rules for this
      element.'

      <basketId> BasketId </basketId> [0..*]
      'A CDS basket identifier'

      <basketId> BasketId </basketId> [1..*]
      'A CDS basket identifier'

   End Choice
End Group: BasketIdentifier.model
<basketCurrency> Currency </basketCurrency> [0..1]
'Specifies the currency for this basket.'

</basket>
```

Diagram





Schema Component Representation

```
<xsd:element name="basket" type=" Basket " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

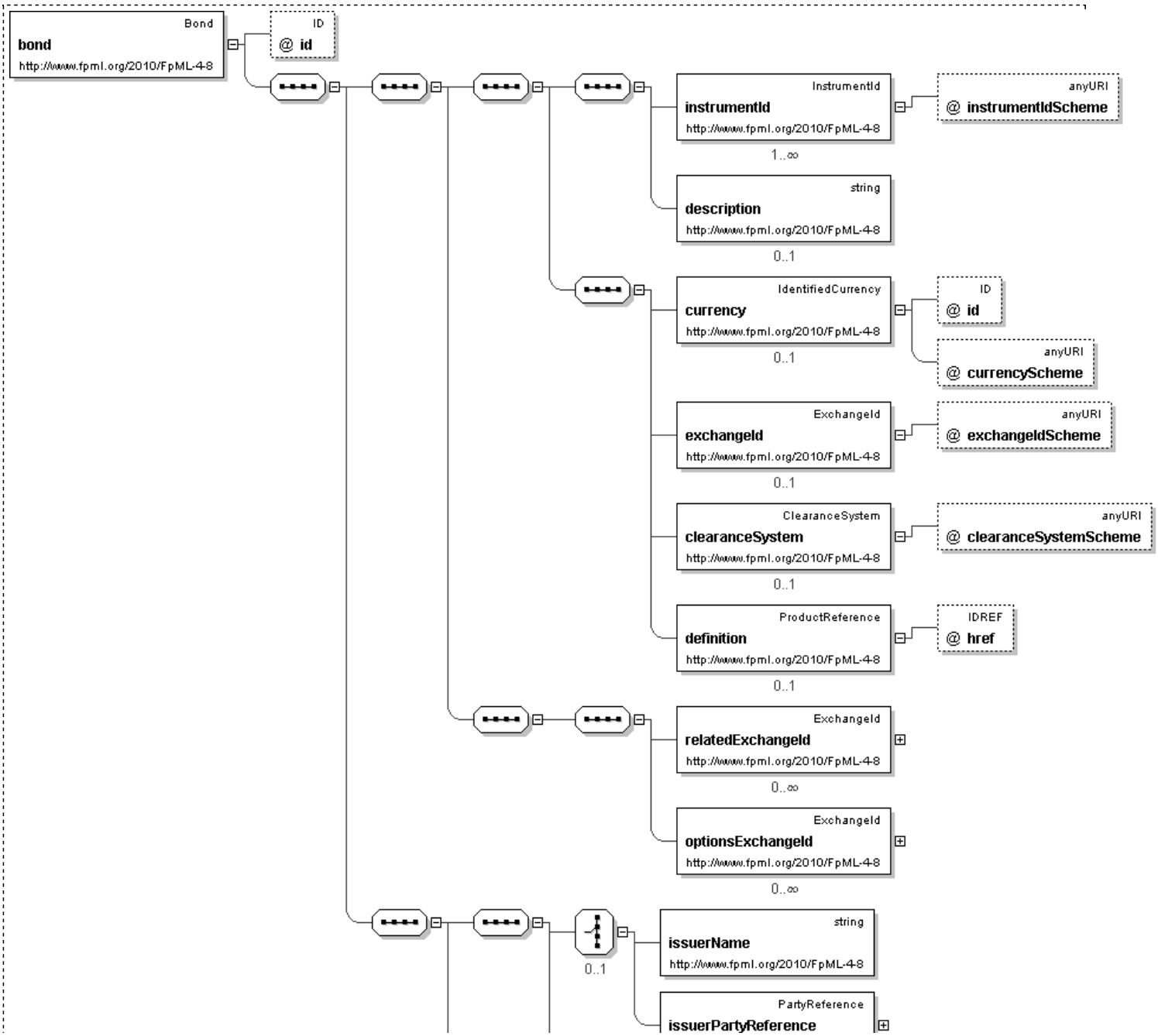
Element: bond

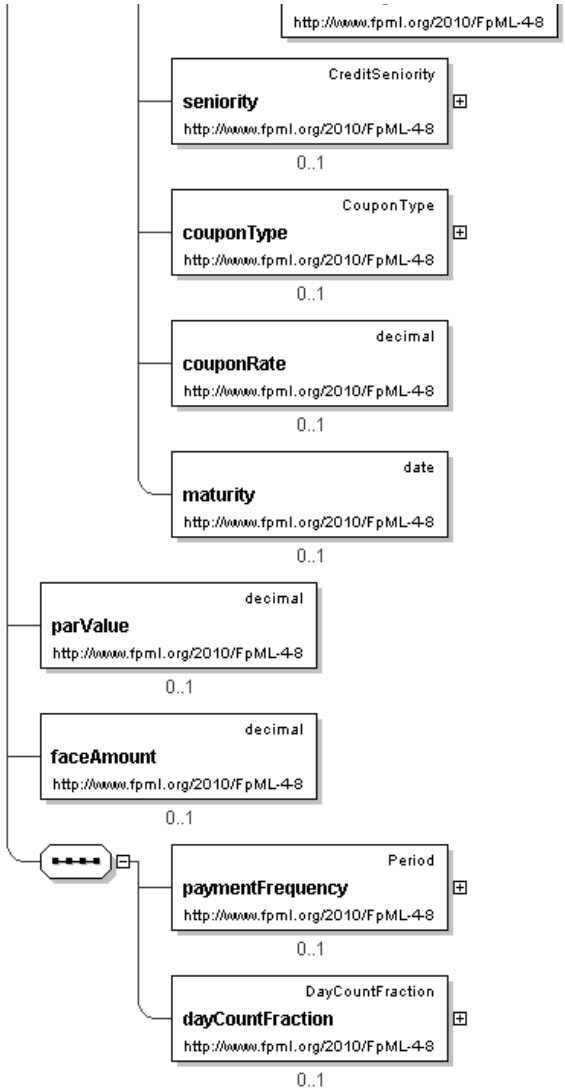
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	bond
Used by (from the same schema document)	Model Group BondChoice.model
Type	Bond
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a bond.

Logical Diagram



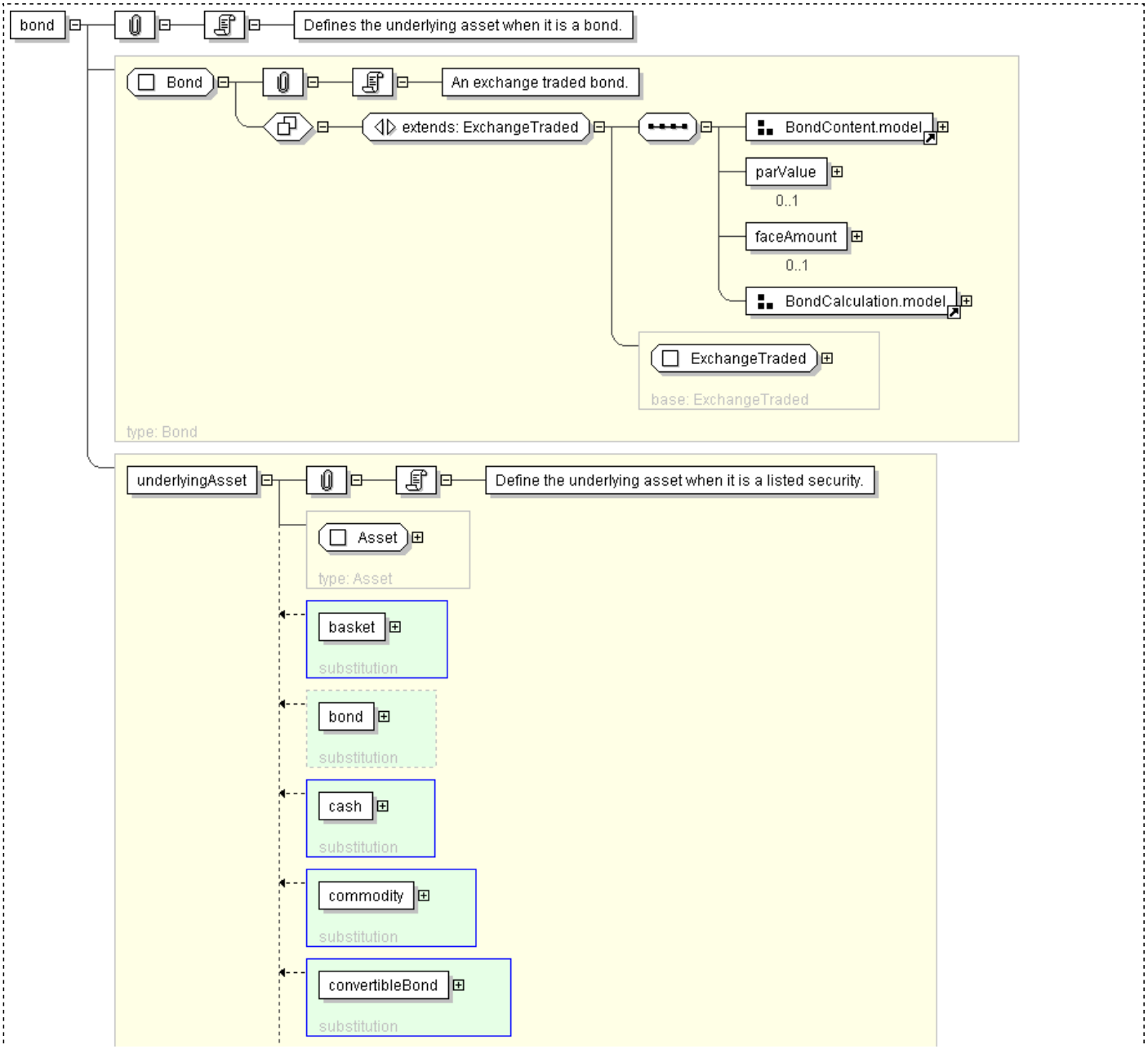


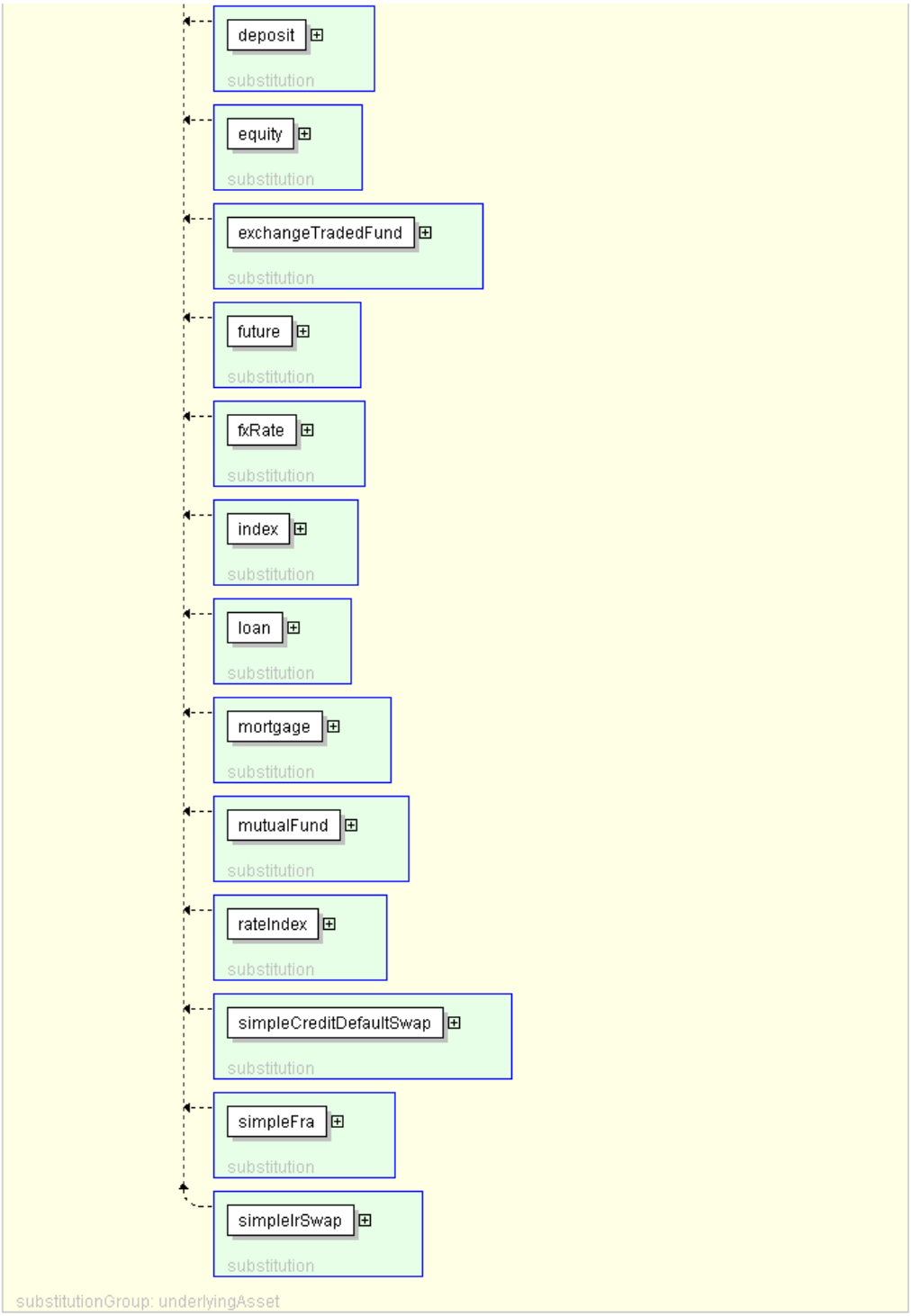
XML Instance Representation

```
<bond
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a
   contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002
   Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
   inconsistency between the terms of the simple product and those of the detailed definition, the values in
   the simple product override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
   'A short form unique identifier for a related exchange. If the element is not present then the exchange
   shall be the primary exchange on which listed futures and options on the underlying are listed. The term
   \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
   'A short form unique identifier for an exchange on which the reference option contract is listed. This is
   to address the case where the reference exchange for the future is different than the one for the option.
   The options Exchange is referenced on share options when Merger Elections are selected as Options Exchange
   Adjustment.'

   Start Choice [0..1]
'Specifies the issuer name of a fixed income security or convertible bond. This name can either be explicitly
stated, or specified as an href into another element of the document, such as the obligor'
```

```
    <issuerName> xsd:string </issuerName> [1]
    <issuerPartyReference> PartyReference </issuerPartyReference> [1]
  End Choice
  <seniority> CreditSeniority </seniority> [0..1]
  'The repayment precedence of a debt instrument.'

  <couponType> CouponType </couponType> [0..1]
  'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-coupon.'

  <couponRate> xsd:decimal </couponRate> [0..1]
  'Specifies the coupon rate (expressed in percentage) of a fixed income security or convertible bond.'

  <maturity> xsd:date </maturity> [0..1]
  'The date when the principal amount of a security becomes due and payable.'

  <parValue> xsd:decimal </parValue> [0..1]
  'Specifies the nominal amount of a fixed income security or convertible bond.'

  <faceAmount> xsd:decimal </faceAmount> [0..1]
  'Specifies the total amount of the issue. Corresponds to the par value multiplied by the number of issued
  security.'

  <paymentFrequency> Period </paymentFrequency> [0..1]
  'Specifies the frequency at which the bond pays, e.g. 6M.'

  <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
  'The day count basis for the bond.'

</bond>
```

Diagram





Schema Component Representation

```
<xsd:element name="bond" type=" Bond " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

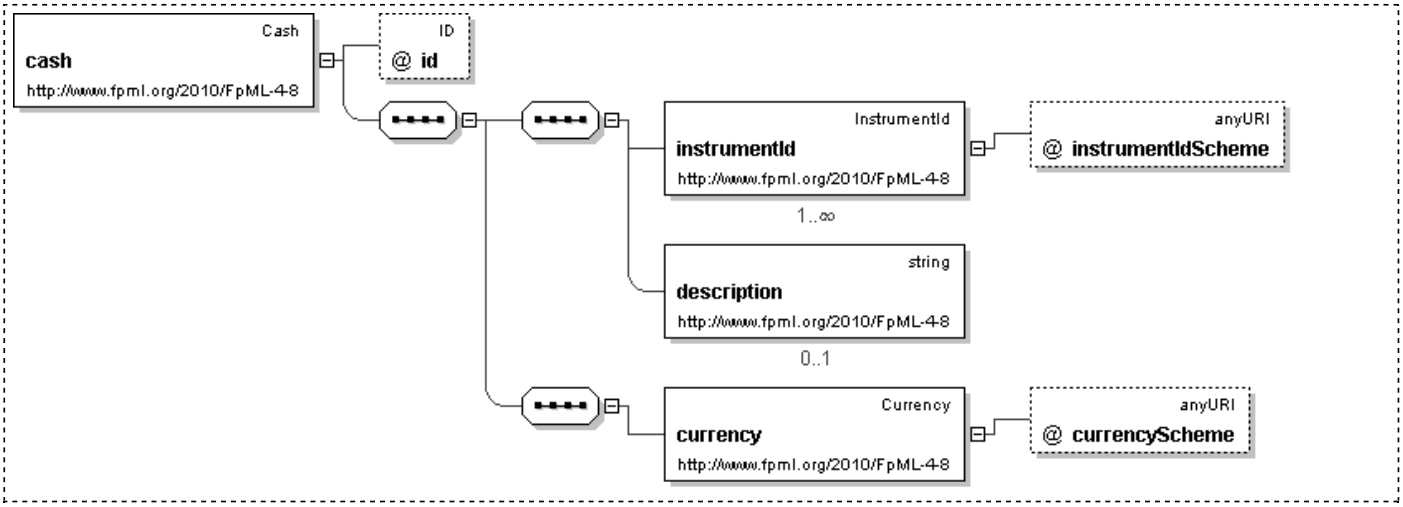
Element: cash

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	cash
Type	Cash
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset type that is a cash payment. Used for specifying discounting factors for future cash flows in the pricing and risk model.

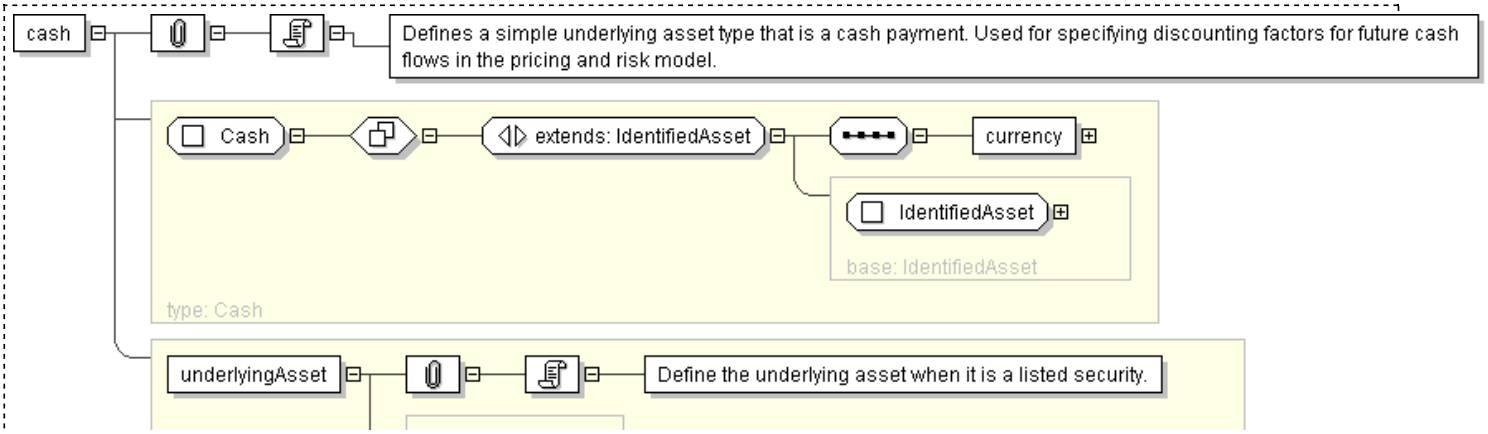
Logical Diagram

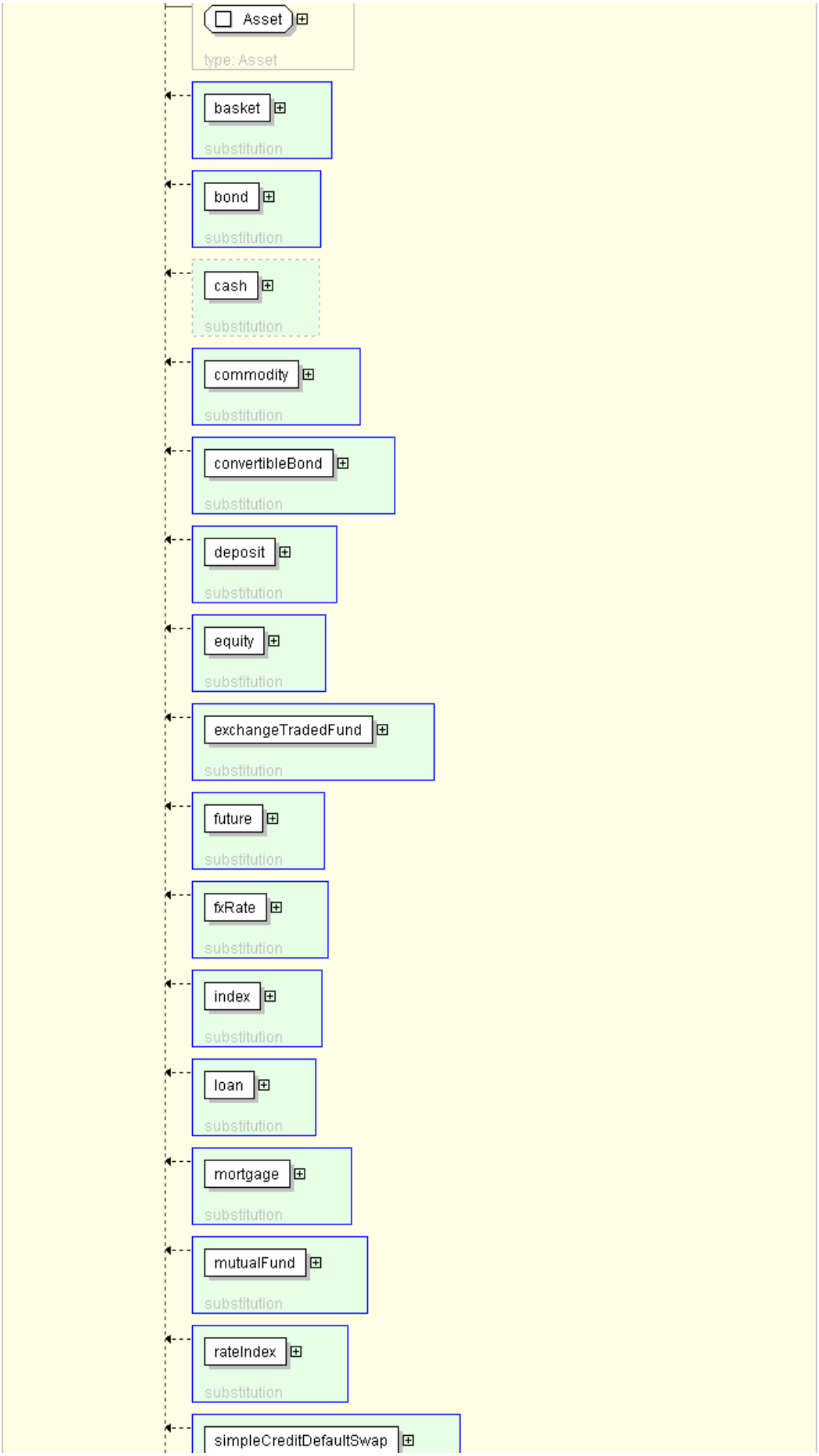


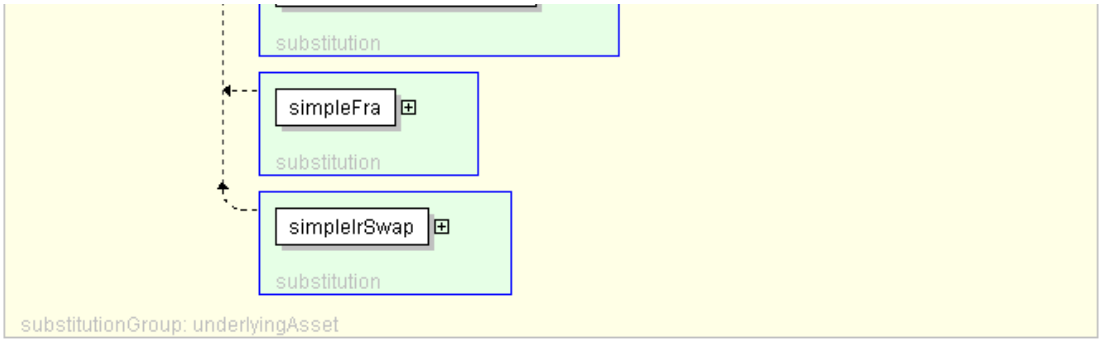
XML Instance Representation

```
<cash
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

</cash>
```

Diagram




Asset
type: Asset
basket
substitution
bond
substitution
cash
substitution
commodity
substitution
convertibleBond
substitution
deposit
substitution
equity
substitution
exchangeTradedFund
substitution
future
substitution
fxRate
substitution
index
substitution
loan
substitution
mortgage
substitution
mutualFund
substitution
rateIndex
substitution
simpleCreditDefaultSwap



Schema Component Representation

```
<xsd:element name="cash" type=" Cash " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

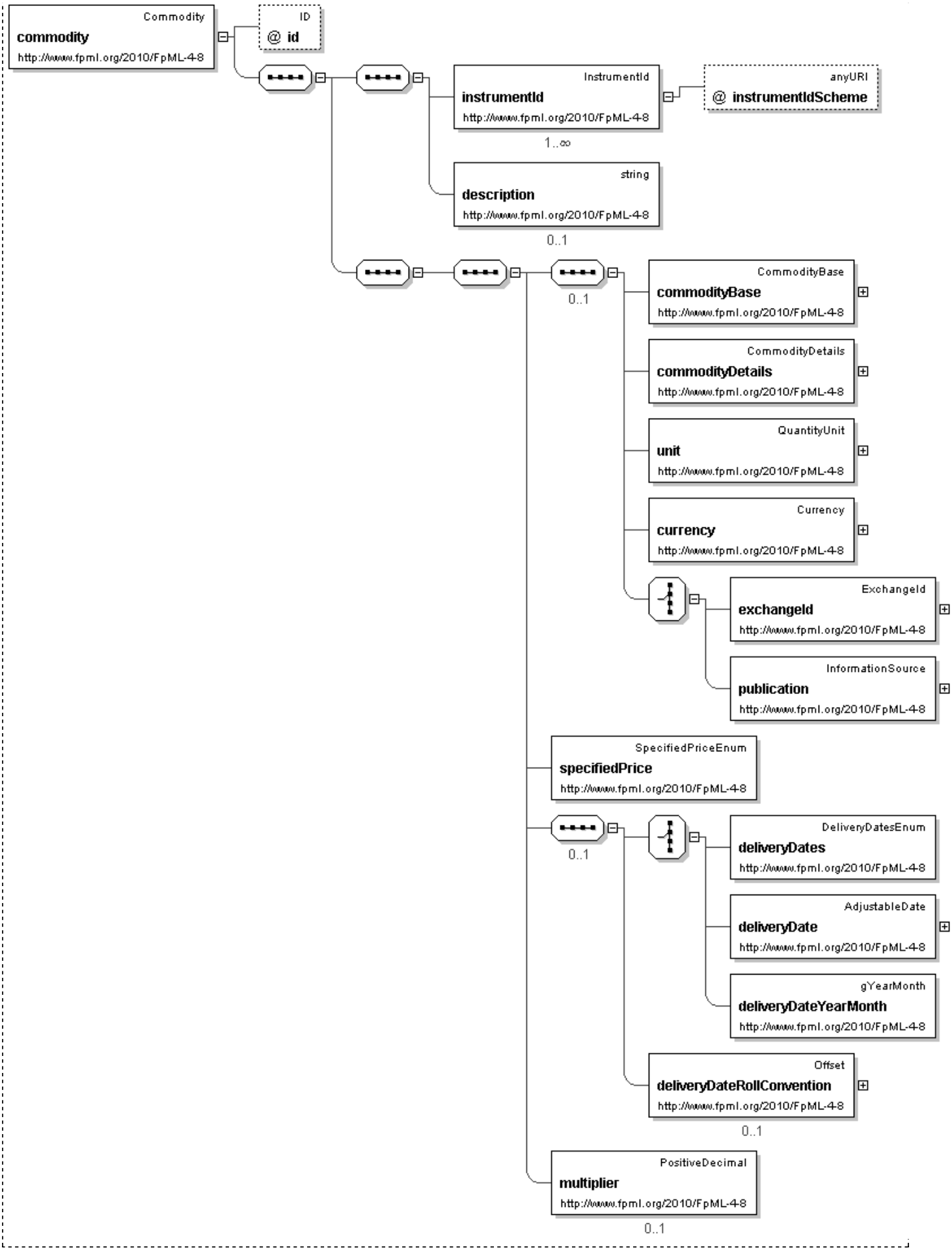
Element: commodity

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	commodity
Type	Commodity
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a commodity.

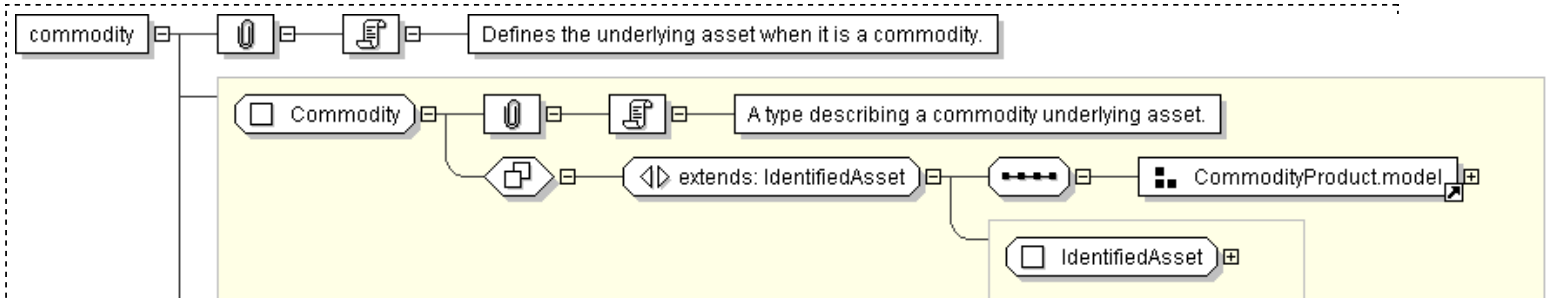
Logical Diagram

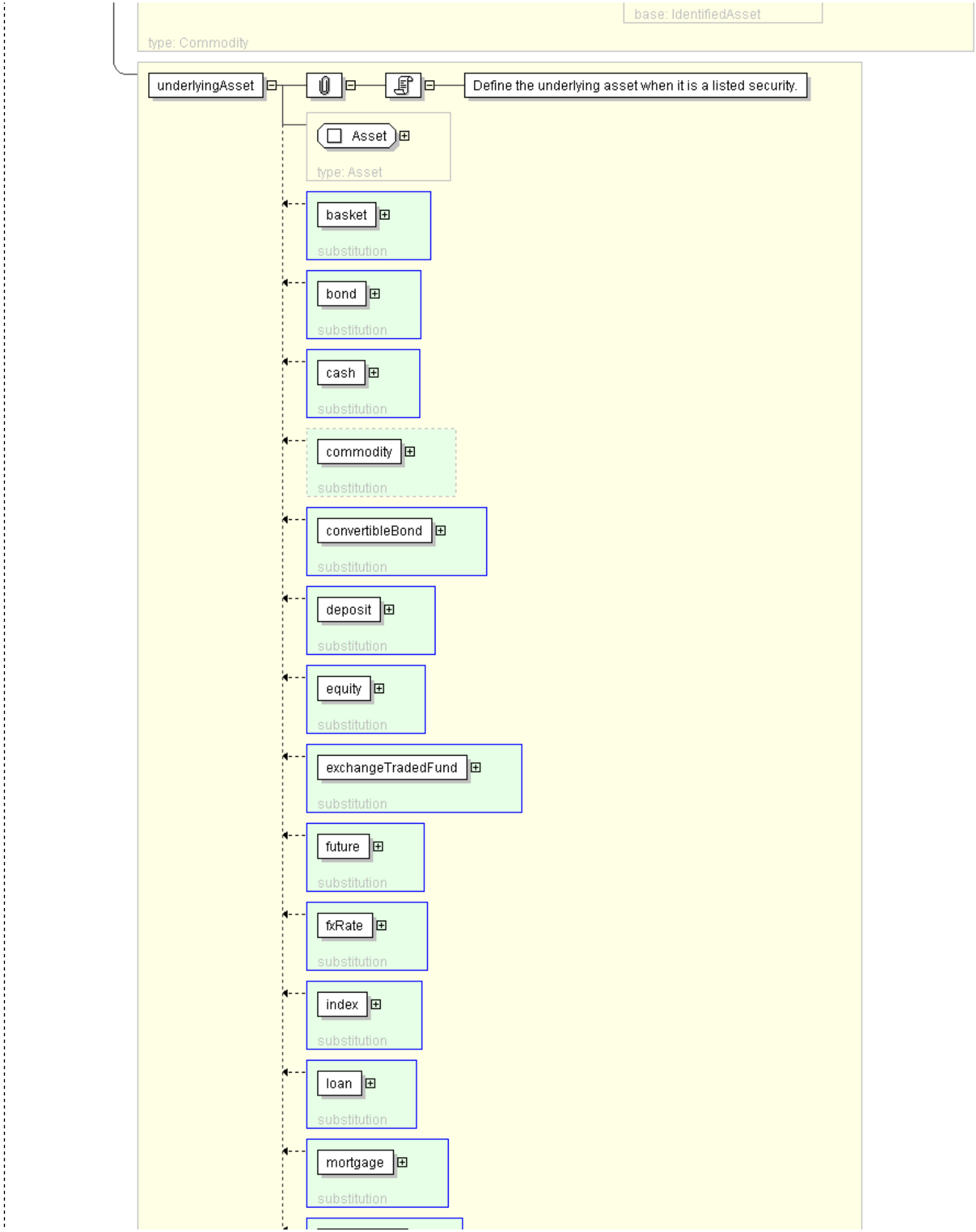

Commodity
commodity
http://www.fpml.org/2010/FpML-4-8
ID
@ id
InstrumentId
instrumentId
http://www.fpml.org/2010/FpML-4-8
1..∞
anyURI
@ instrumentIdScheme
string
description
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
CommodityBase
commodityBase
http://www.fpml.org/2010/FpML-4-8
CommodityDetails
commodityDetails
http://www.fpml.org/2010/FpML-4-8
QuantityUnit
unit
http://www.fpml.org/2010/FpML-4-8
Currency
currency
http://www.fpml.org/2010/FpML-4-8
ExchangeId
exchangeId
http://www.fpml.org/2010/FpML-4-8
InformationSource
publication
http://www.fpml.org/2010/FpML-4-8
SpecifiedPriceEnum
specifiedPrice
http://www.fpml.org/2010/FpML-4-8
0..1
DeliveryDatesEnum
deliveryDates
http://www.fpml.org/2010/FpML-4-8
AdjustableDate
deliveryDate
http://www.fpml.org/2010/FpML-4-8
gYearMonth
deliveryDateYearMonth
http://www.fpml.org/2010/FpML-4-8
Offset
deliveryDateRollConvention
http://www.fpml.org/2010/FpML-4-8
0..1
PositiveDecimal
multiplier
http://www.fpml.org/2010/FpML-4-8
0..1

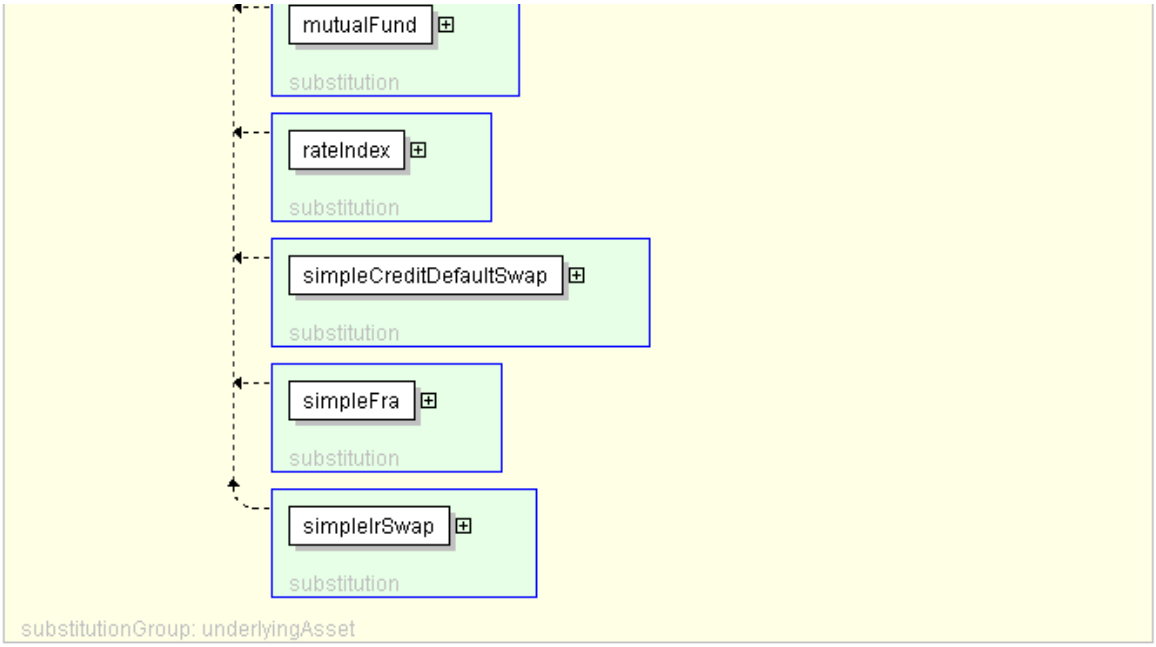
XML Instance Representation

```
<commodity
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   Start Group: CommodityReferencePriceFramework.model [0..1]
      <commodityBase> CommodityBase </commodityBase> [1]
       'A coding scheme value to identify the base type of the commodity being traded. Where
       possible, this should follow the naming convention used in the 2005 ISDA Commodity
       Definitions. For example, \'Oil\'.'

      <commodityDetails> CommodityDetails </commodityDetails> [1]
       'A coding scheme value to identify the commodity being traded more specifically. Where
       possible, this should follow the naming convention used in the 2005 ISDA Commodity
       Definitions. For example, \'Brent\'.'

      <unit> QuantityUnit </unit> [1]
       'A coding scheme value to identify the unit in which the undelryer is denominated. Where
       possible, this should follow the naming convention used in the 2005 ISDA Commodity
       Definitions.'

      <currency> Currency </currency> [1]
       'The currency in which the Commodity Reference Price is published.'

      Start Choice [1]
         <exchangeId> ExchangeId </exchangeId> [1]
          'For those commodities being traded with reference to the price of a listed future, the
          exchange where that future is listed should be specified here.'

         <publication> InformationSource </publication> [1]
          'For those commodities being traded with reference to a price distributed by a
          publication, that publication should be specified here.'

      End Choice
   End Group: CommodityReferencePriceFramework.model
   <specifiedPrice> SpecifiedPriceEnum </specifiedPrice> [1]
    'The Specified Price is not defined in the Commodity Reference Price and so needs to be
    stated in the Underlyer definition as it will impact the calculation of the Floating Price.'

   Start Sequence [0..1]
      Start Choice [1]
         <deliveryDates> DeliveryDatesEnum </deliveryDates> [1]
          'The Delivery Date is a NearbyMonth, for use when the Commodity Transaction references
          Futures Contract.'

         <deliveryDate> AdjustableDate </deliveryDate> [1]
          'The Delivery Date is a fixed, single day.'

         <deliveryDateYearMonth> xsd:gYearMonth </deliveryDateYearMonth> [1]
          'The Delivery Date is a fixed, single month.'

      End Choice
      <deliveryDateRollConvention> Offset </deliveryDateRollConvention> [0..1]
       'Specifies, for a Commodity Transaction that references a listed future via the
       deliveryDates element, the day on which the specified future will roll to the next nearby
       month when the referenced future expires. If the future will not roll at all - i.e. the
       price will be taken from the expiring contract, 0 should be specified here. If the future
       will roll to the next nearby on the last trading day - i.e. the price will be taken from
       the next nearby on the last trading day, then 1 should be specified and so on.'

   End Sequence
   <multiplier> PositiveDecimal </multiplier> [0..1]
    'Specifies the multiplier associated with a Transaction.'

</commodity>
```

Diagram




base: IdentifiedAsset
type: Commodity
underlyingAsset
Define the underlying asset when it is a listed security.
Asset
type: Asset
basket
substitution
bond
substitution
cash
substitution
commodity
substitution
convertibleBond
substitution
deposit
substitution
equity
substitution
exchangeTradedFund
substitution
future
substitution
fxRate
substitution
index
substitution
loan
substitution
mortgage
substitution



Schema Component Representation

```
<xsd:element name="commodity" type=" Commodity " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

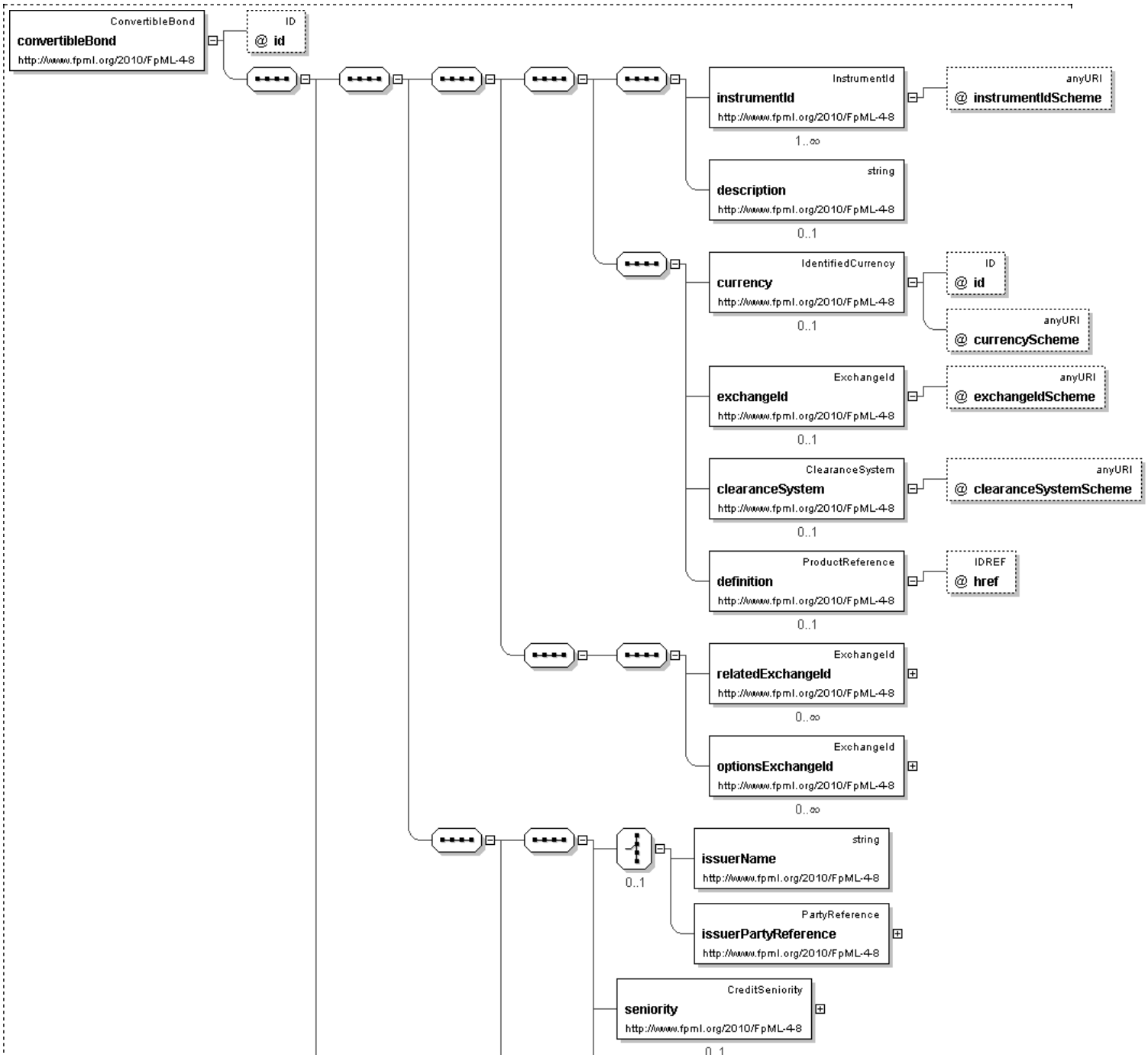
Element: convertibleBond

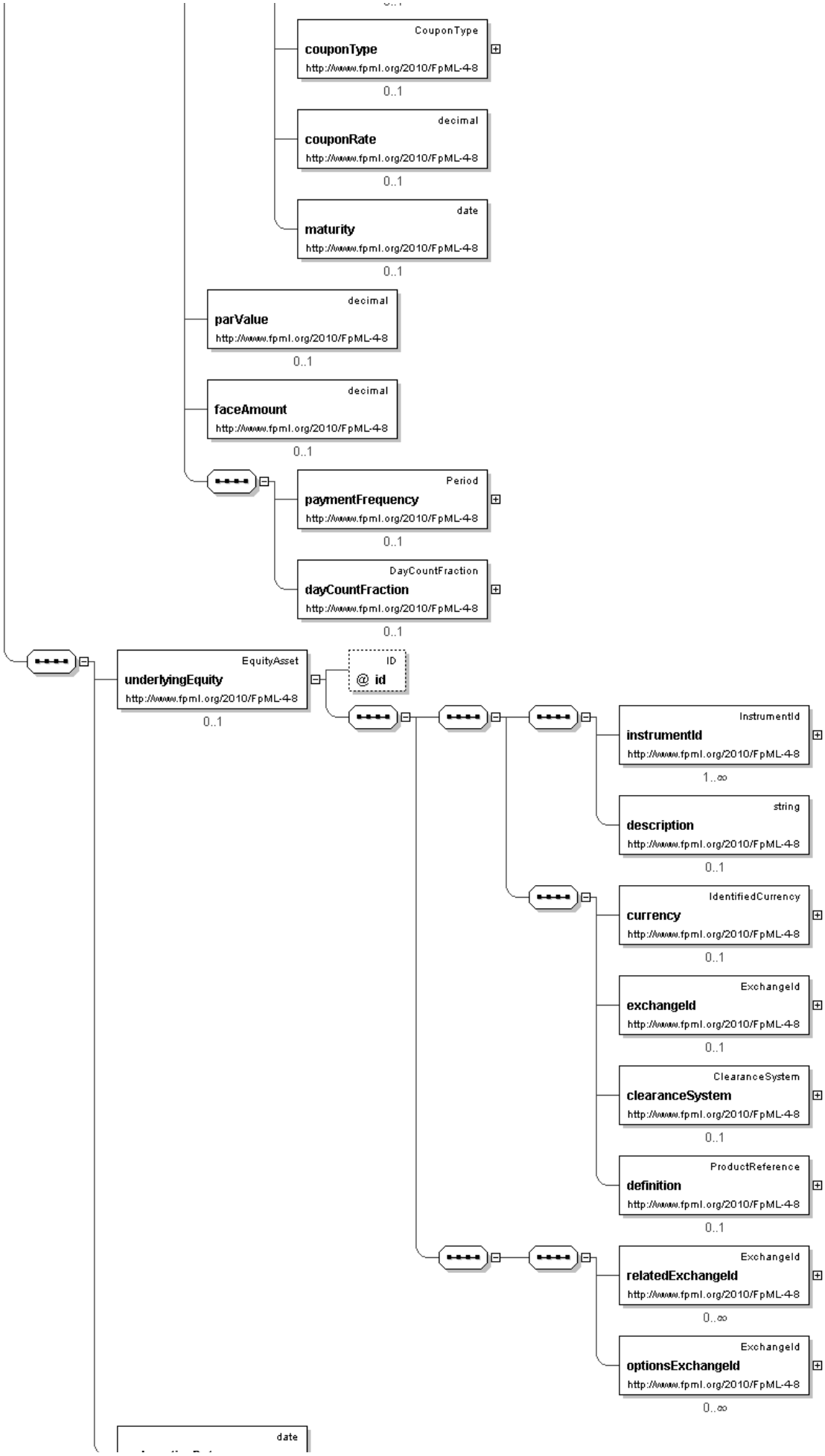
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	convertibleBond
Used by (from the same schema document)	Model Group BondChoice.model
Type	ConvertibleBond
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a convertible bond.

Logical Diagram

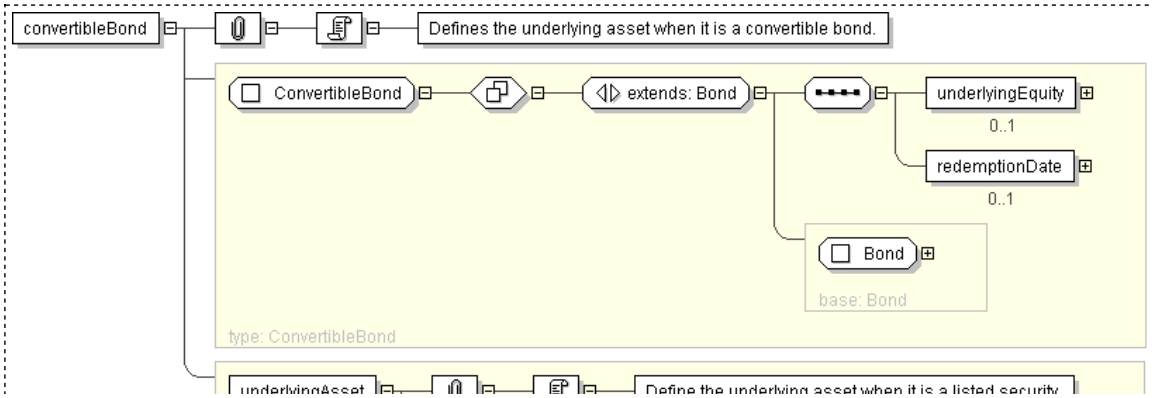


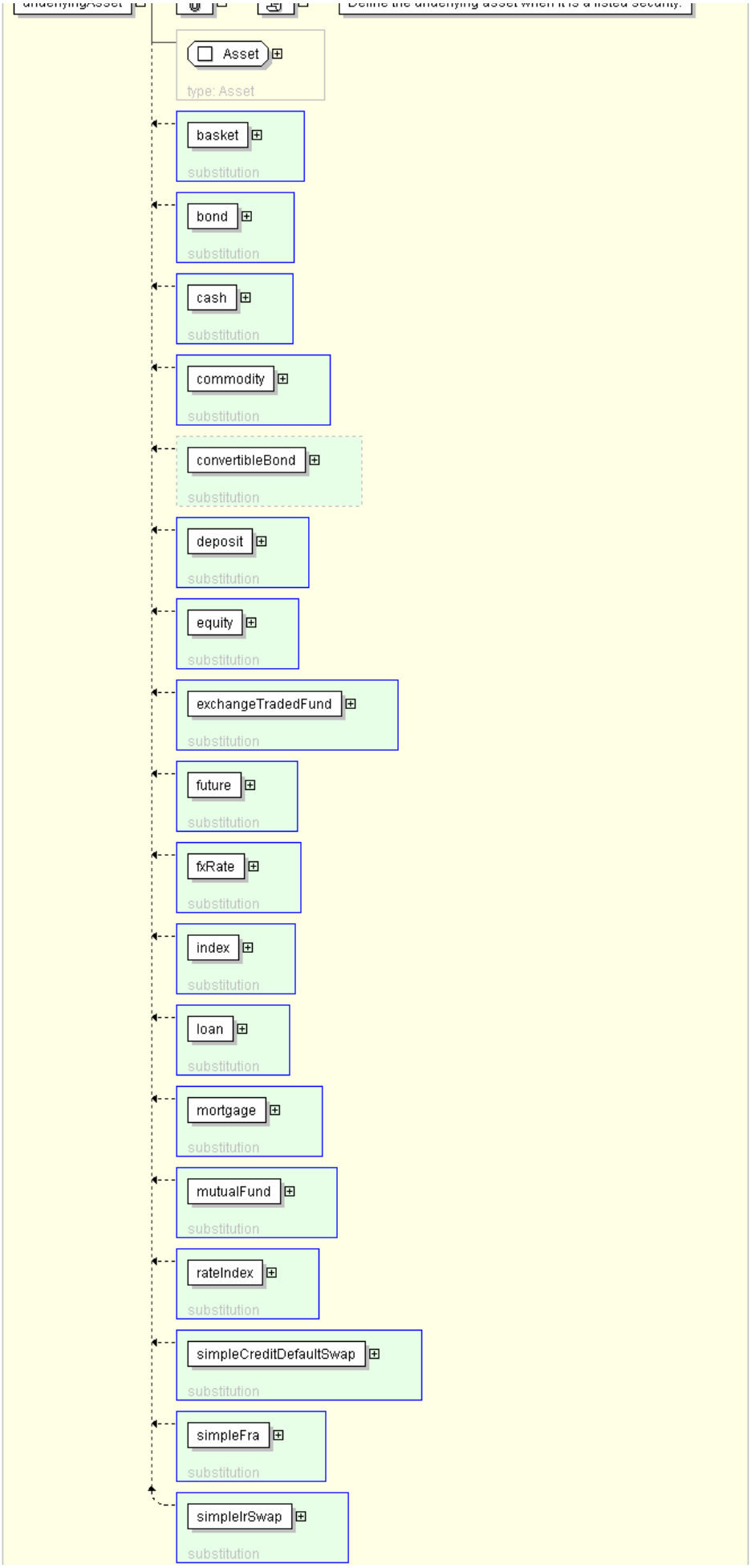

CouponType
couponType
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
couponRate
http://www.fpml.org/2010/FpML-4-8
0..1
date
maturity
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
parValue
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
faceAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Period
paymentFrequency
http://www.fpml.org/2010/FpML-4-8
0..1
DayCountFraction
dayCountFraction
http://www.fpml.org/2010/FpML-4-8
0..1
EquityAsset
underlyingEquity
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
InstrumentId
instrumentId
http://www.fpml.org/2010/FpML-4-8
1..∞
string
description
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedCurrency
currency
http://www.fpml.org/2010/FpML-4-8
0..1
ExchangeId
exchangeId
http://www.fpml.org/2010/FpML-4-8
0..1
ClearanceSystem
clearanceSystem
http://www.fpml.org/2010/FpML-4-8
0..1
ProductReference
definition
http://www.fpml.org/2010/FpML-4-8
0..1
ExchangeId
relatedExchangeId
http://www.fpml.org/2010/FpML-4-8
0..∞
ExchangeId
optionsExchangeId
http://www.fpml.org/2010/FpML-4-8
0..∞
date

XML Instance Representation

```
<convertibleBond
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural
   payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives
   Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
   inconsistency between the terms of the simple product and those of the detailed definition, the values in the simple
   product override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
   'A short form unique identifier for a related exchange. If the element is not present then the exchange shall be the
   primary exchange on which listed futures and options on the underlying are listed. The term \"Exchange\" is assumed
   to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
   'A short form unique identifier for an exchange on which the reference option contract is listed. This is to address
   the case where the reference exchange for the future is different than the one for the option. The options Exchange
   is referenced on share options when Merger Elections are selected as Options Exchange Adjustment.'

   Start Choice [0..1]
'Specifies the issuer name of a fixed income security or convertible bond. This name can either be explicitly stated, or
specified as an href into another element of the document, such as the obligor'


      <issuerName> xsd:string </issuerName> [1]
      <issuerPartyReference> PartyReference </issuerPartyReference> [1]
   End Choice
   <seniority> CreditSeniority </seniority> [0..1]
   'The repayment precedence of a debt instrument.'

   <couponType> CouponType </couponType> [0..1]
   'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-coupon.'

   <couponRate> xsd:decimal </couponRate> [0..1]
   'Specifies the coupon rate (expressed in percentage) of a fixed income security or convertible bond.'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of a security becomes due and payable.'

   <parValue> xsd:decimal </parValue> [0..1]
   'Specifies the nominal amount of a fixed income security or convertible bond.'

   <faceAmount> xsd:decimal </faceAmount> [0..1]
   'Specifies the total amount of the issue. Corresponds to the par value multiplied by the number of issued security.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
   'Specifies the frequency at which the bond pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
   'The day count basis for the bond.'

   <underlyingEquity> EquityAsset </underlyingEquity> [0..1]
   'Specifies the equity in which the convertible bond can be converted.'

   <redemptionDate> xsd:date </redemptionDate> [0..1]
   'Earlier date between the convertible bond put dates and its maturity date.'

</convertibleBond>
```

Diagram




underlyingAsset
Define the underlying asset when it is a listed security.
Asset
type: Asset
basket
substitution
bond
substitution
cash
substitution
commodity
substitution
convertibleBond
substitution
deposit
substitution
equity
substitution
exchangeTradedFund
substitution
future
substitution
fxRate
substitution
index
substitution
loan
substitution
mortgage
substitution
mutualFund
substitution
rateIndex
substitution
simpleCreditDefaultSwap
substitution
simpleFra
substitution
simpleIrSwap
substitution

substitutionGroup: underlyingAsset

Schema Component Representation

```
<xsd:element name="convertibleBond" type=" ConvertibleBond " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

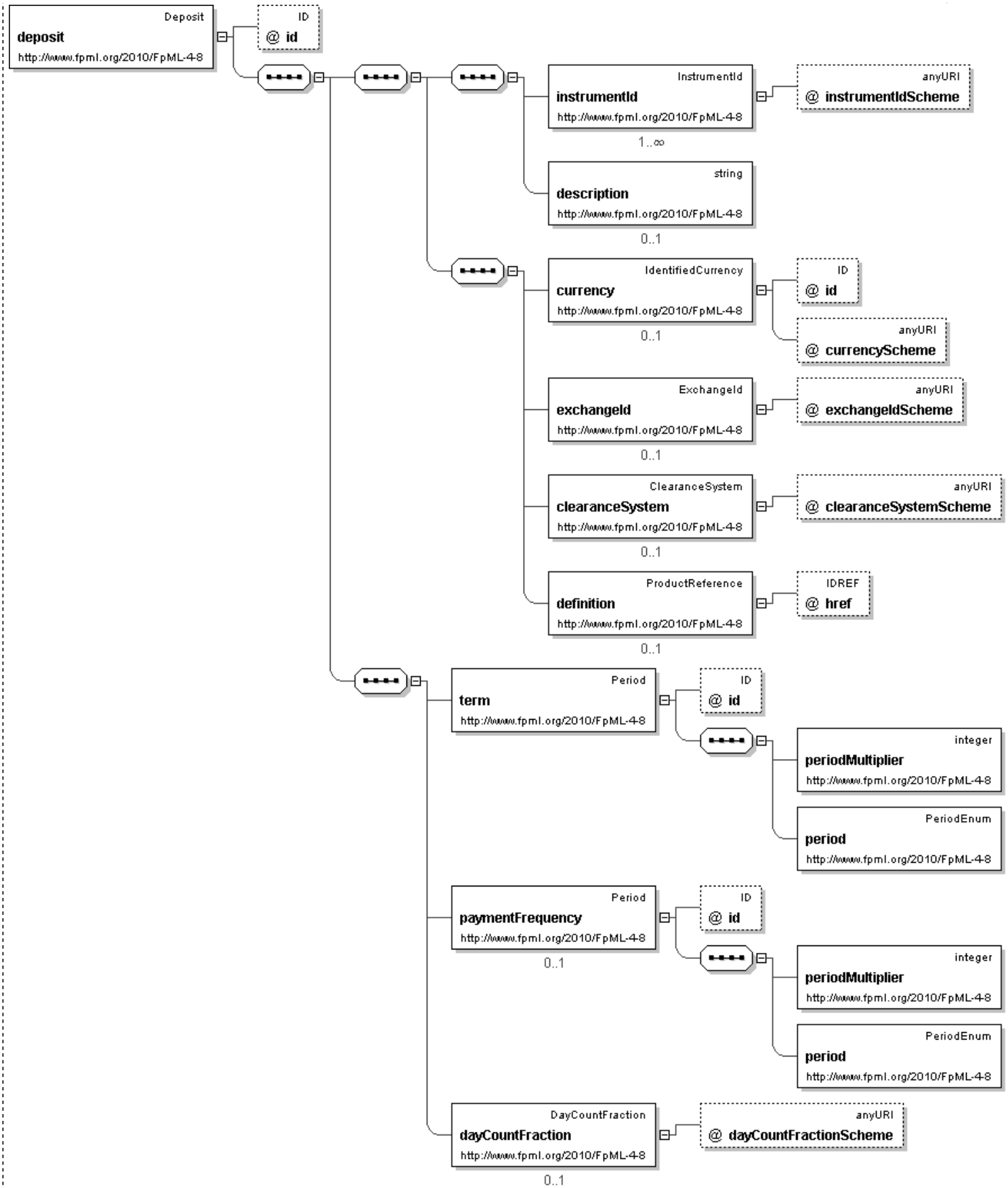
Element: deposit

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	deposit
Type	Deposit
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is a term deposit.

Logical Diagram

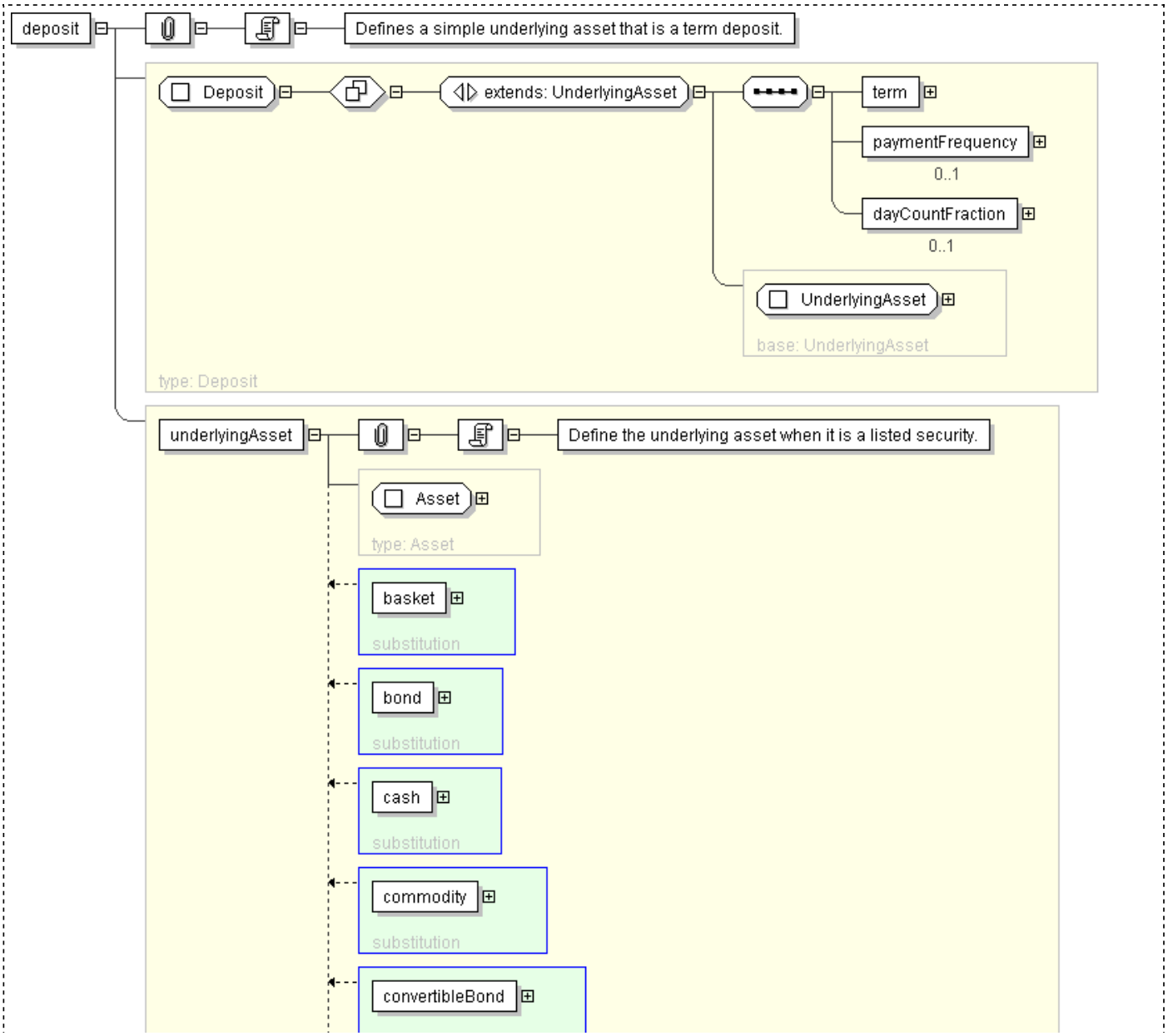


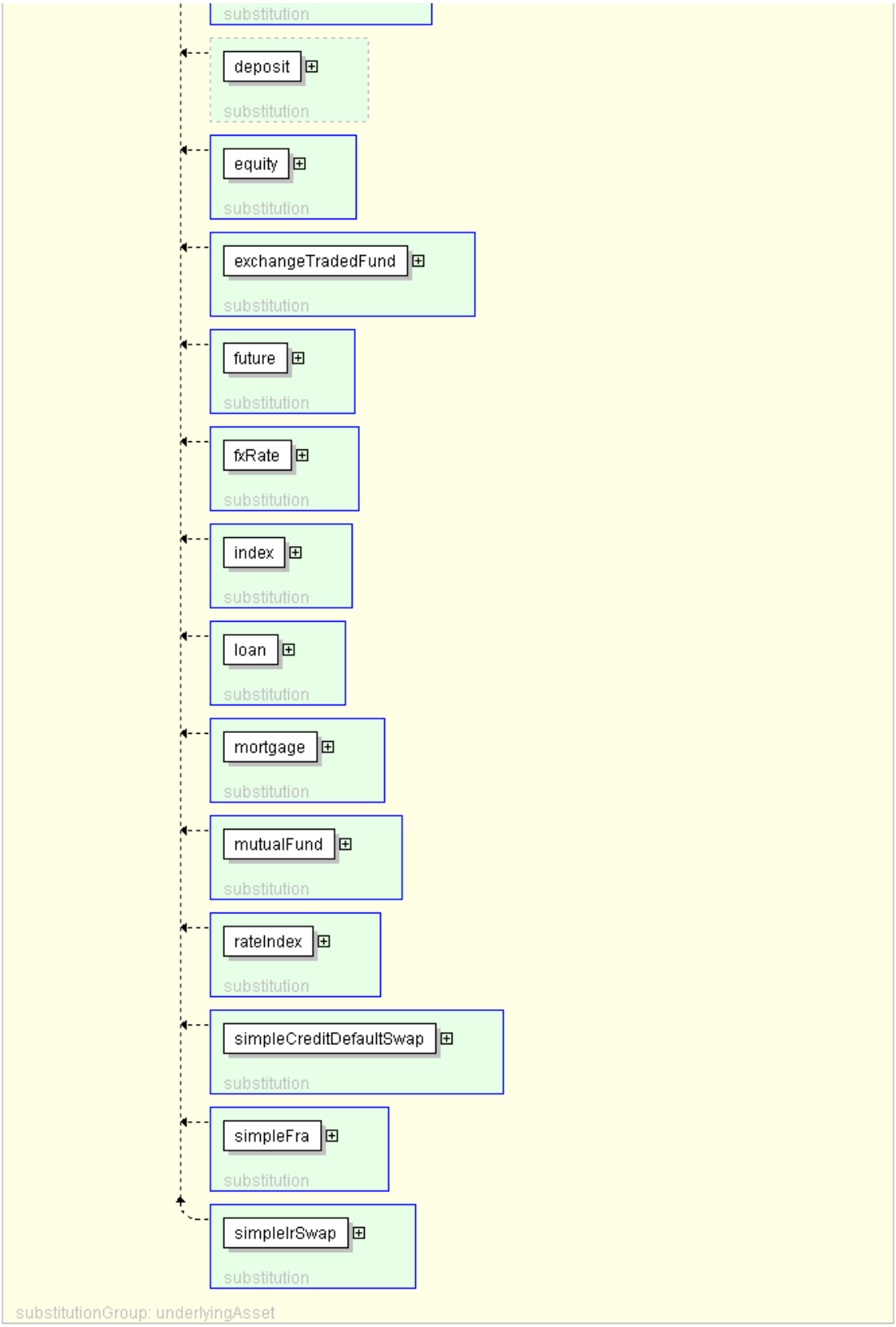
XML Instance Representation

```
<deposit
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'
```

```
    <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating
    a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
    2002 Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail.
    In case of inconsistency between the terms of the simple product and those of the detailed
    definition, the values in the simple product override those in the detailed definition.'

    <term> Period </term> [1]
    'Specifies the term of the deposit, e.g. 5Y.'

    <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the deposit pays, e.g. 6M.'

    <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the deposit.'

</deposit>
```

Diagram





Schema Component Representation

```
<xsd:element name="deposit" type=" Deposit " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

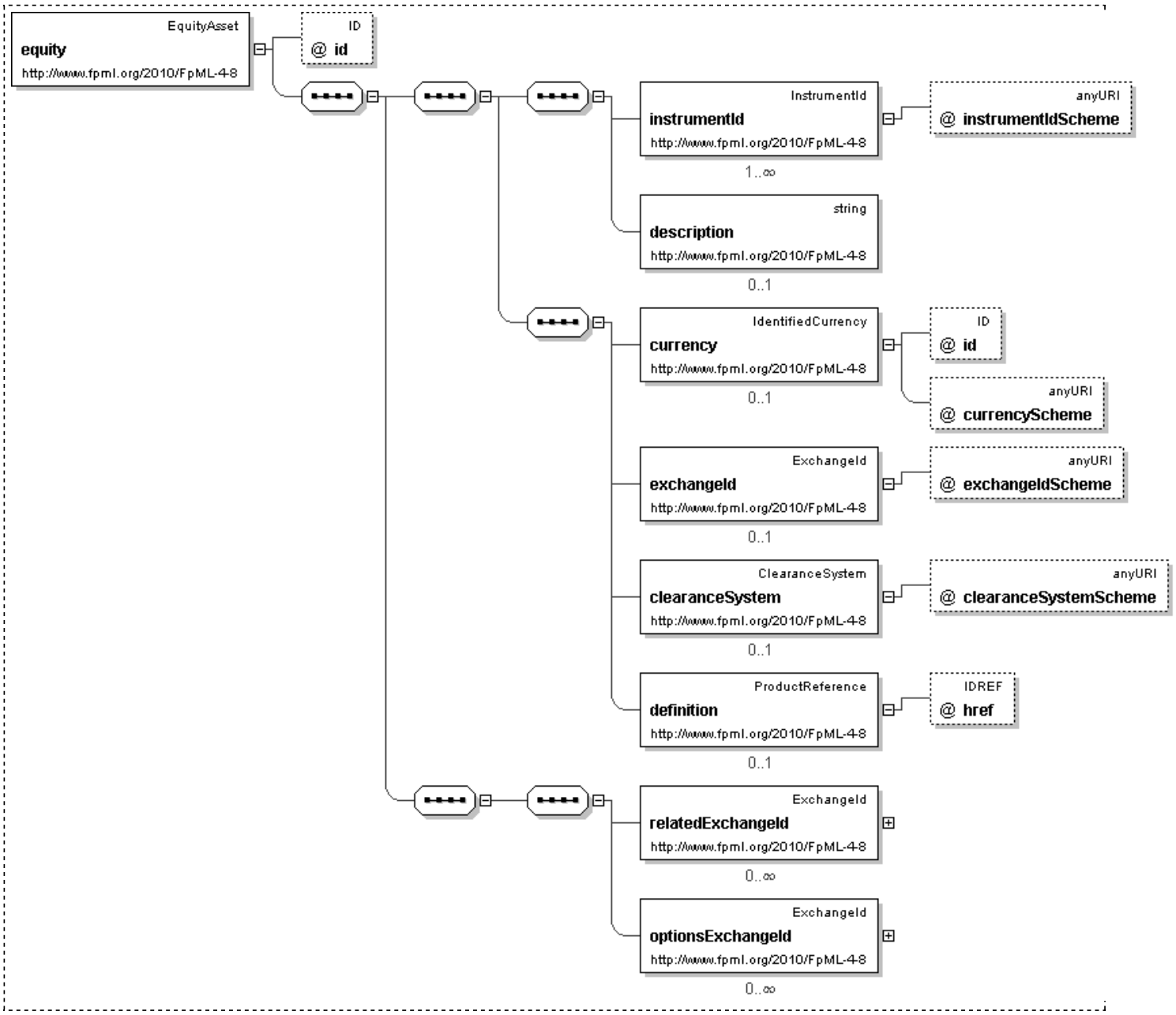
Element: equity

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	equity
Type	EquityAsset
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a listed equity.

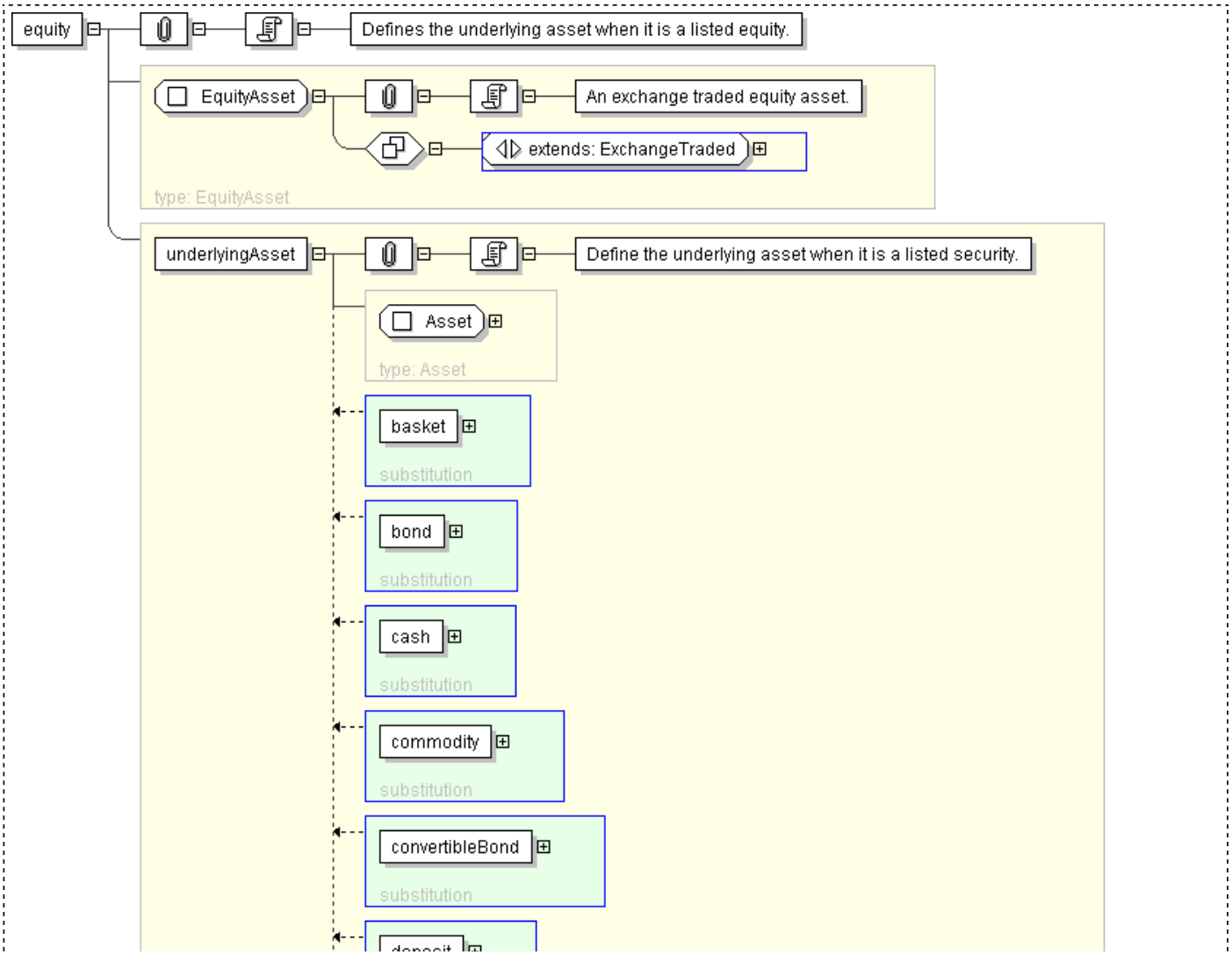
Logical Diagram

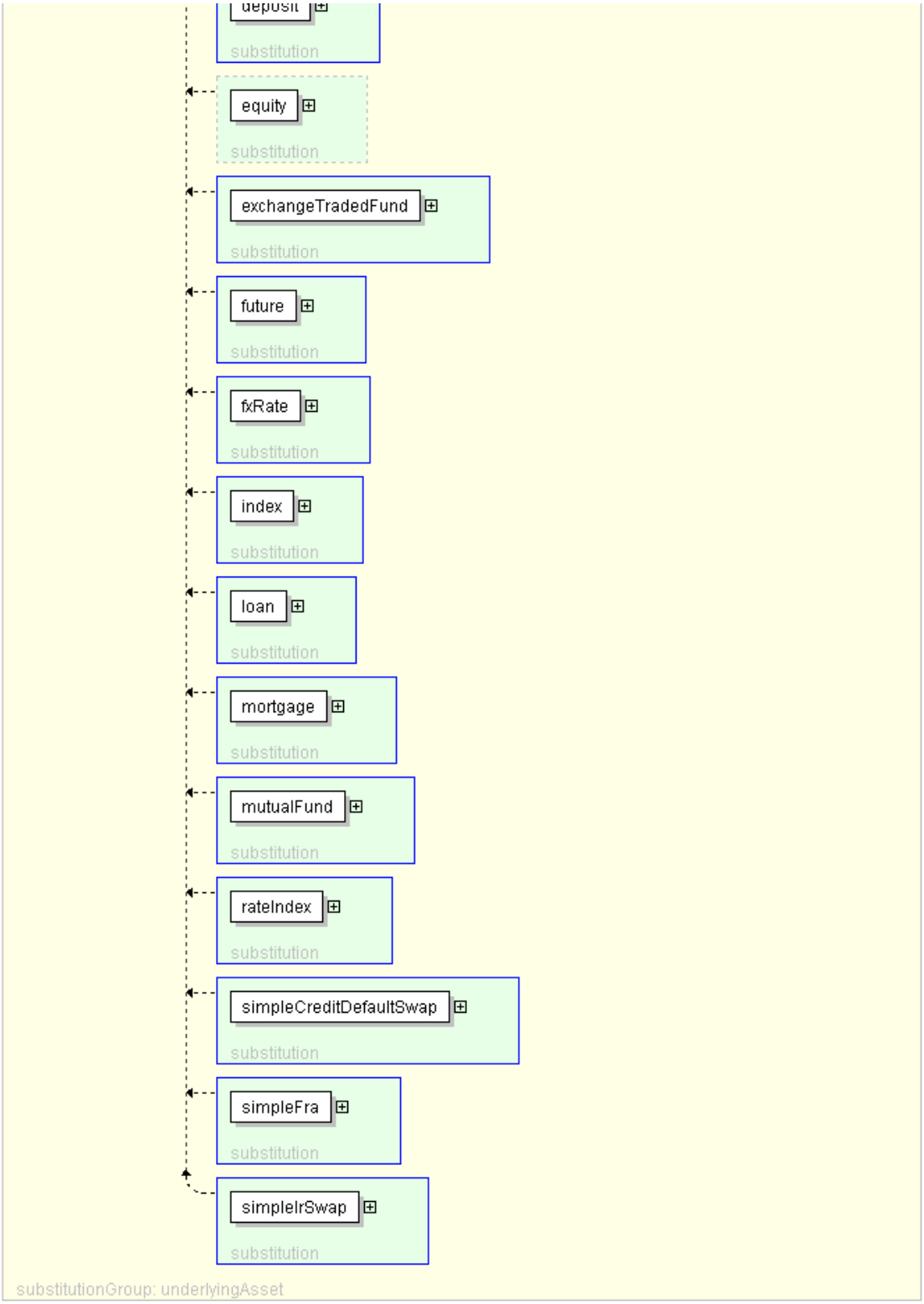


XML Instance Representation

```
<equity
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating
   a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
   2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail.
   In case of inconsistency between the terms of the simple product and those of the detailed
   definition, the values in the simple product override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
   'A short form unique identifier for a related exchange. If the element is not present then the
   exchange shall be the primary exchange on which listed futures and options on the underlying are
   listed. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity
   Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
   'A short form unique identifier for an exchange on which the reference option contract is listed.
   This is to address the case where the reference exchange for the future is different than the one
   for the option. The options Exchange is referenced on share options when Merger Elections are
   selected as Options Exchange Adjustment.'

</equity>
```

Diagram





Schema Component Representation

```
<xsd:element name="equity" type=" EquityAsset " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

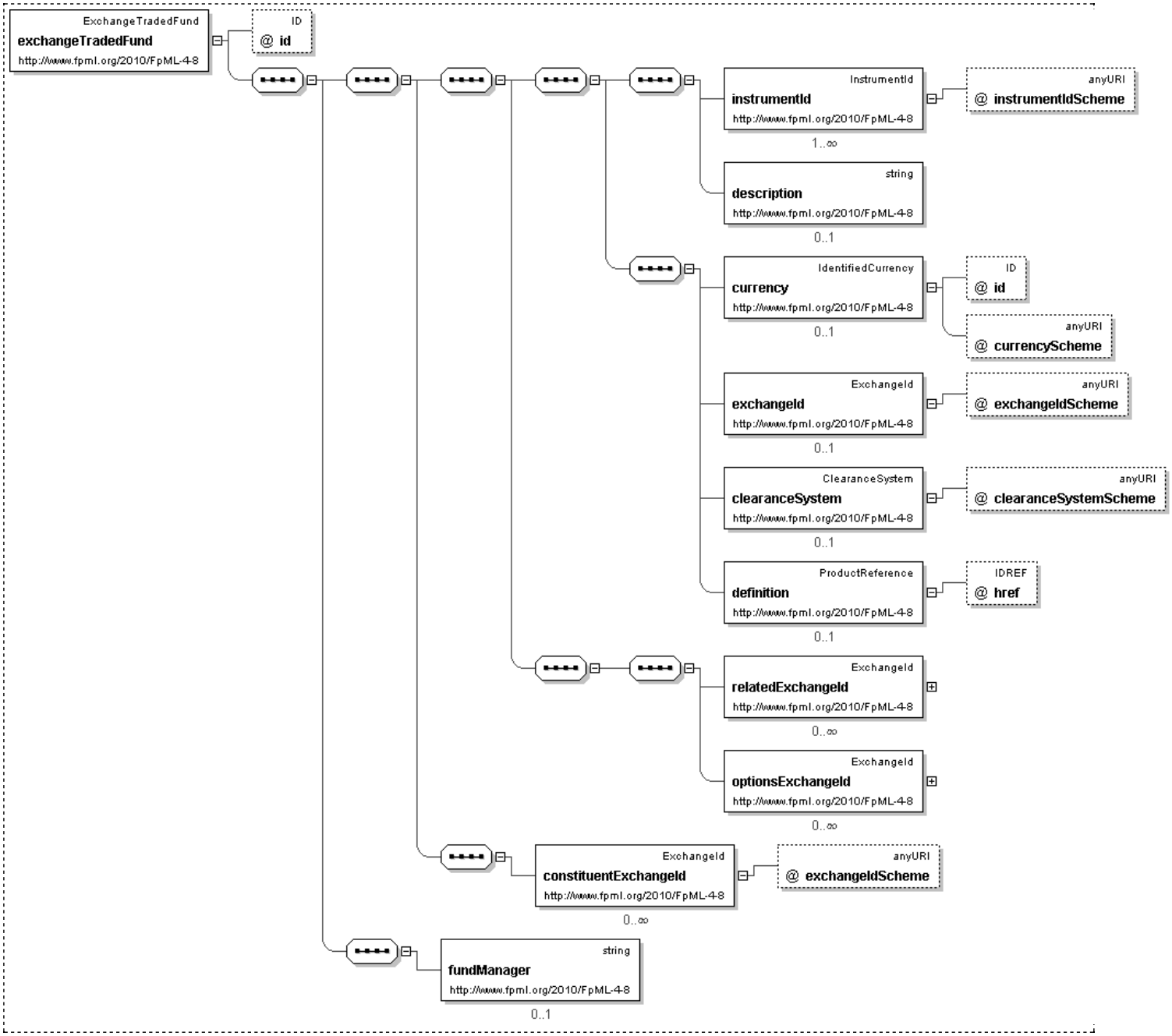
Element: exchangeTradedFund

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	exchangeTradedFund
Type	ExchangeTradedFund
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is an exchange-traded fund.

Logical Diagram

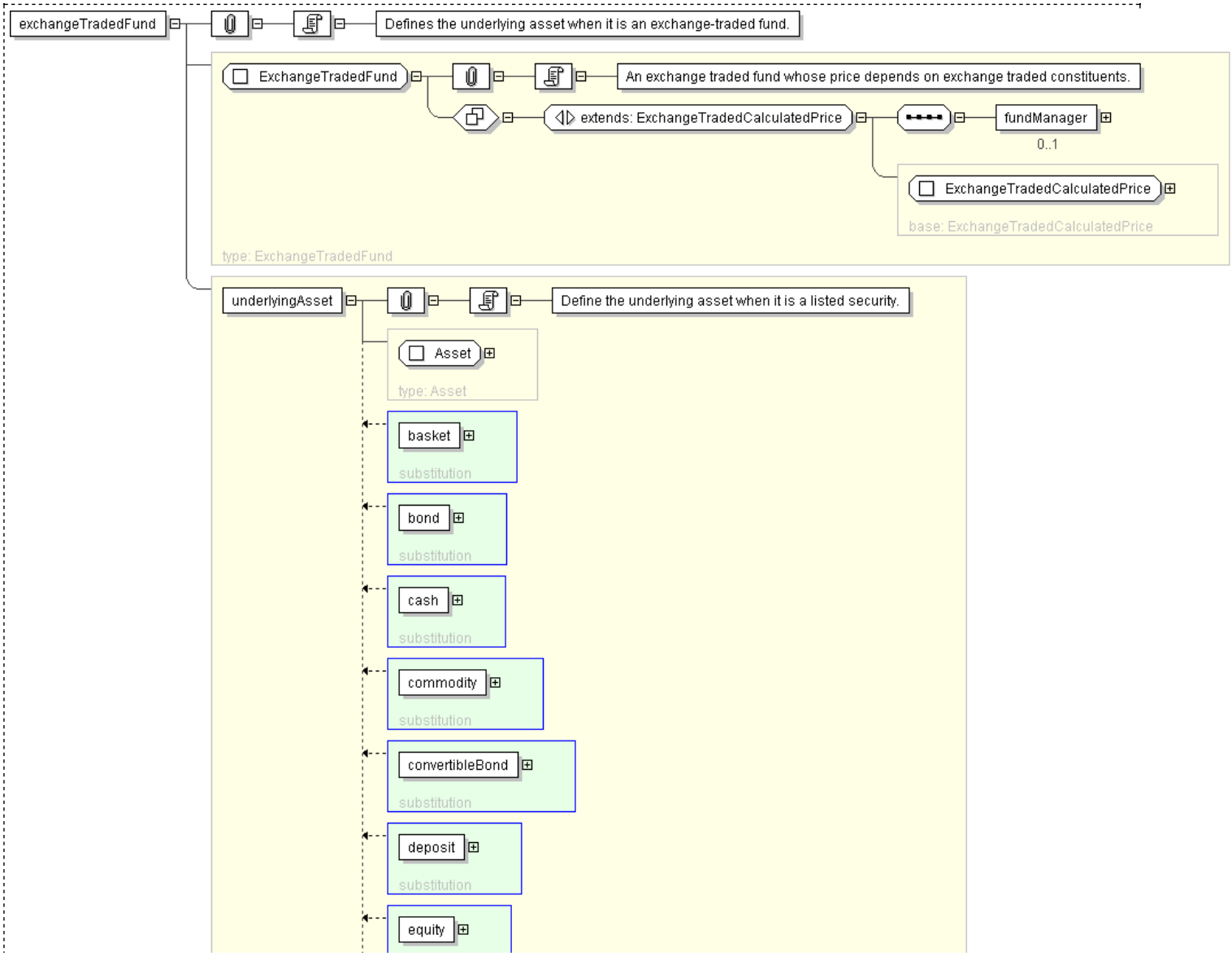


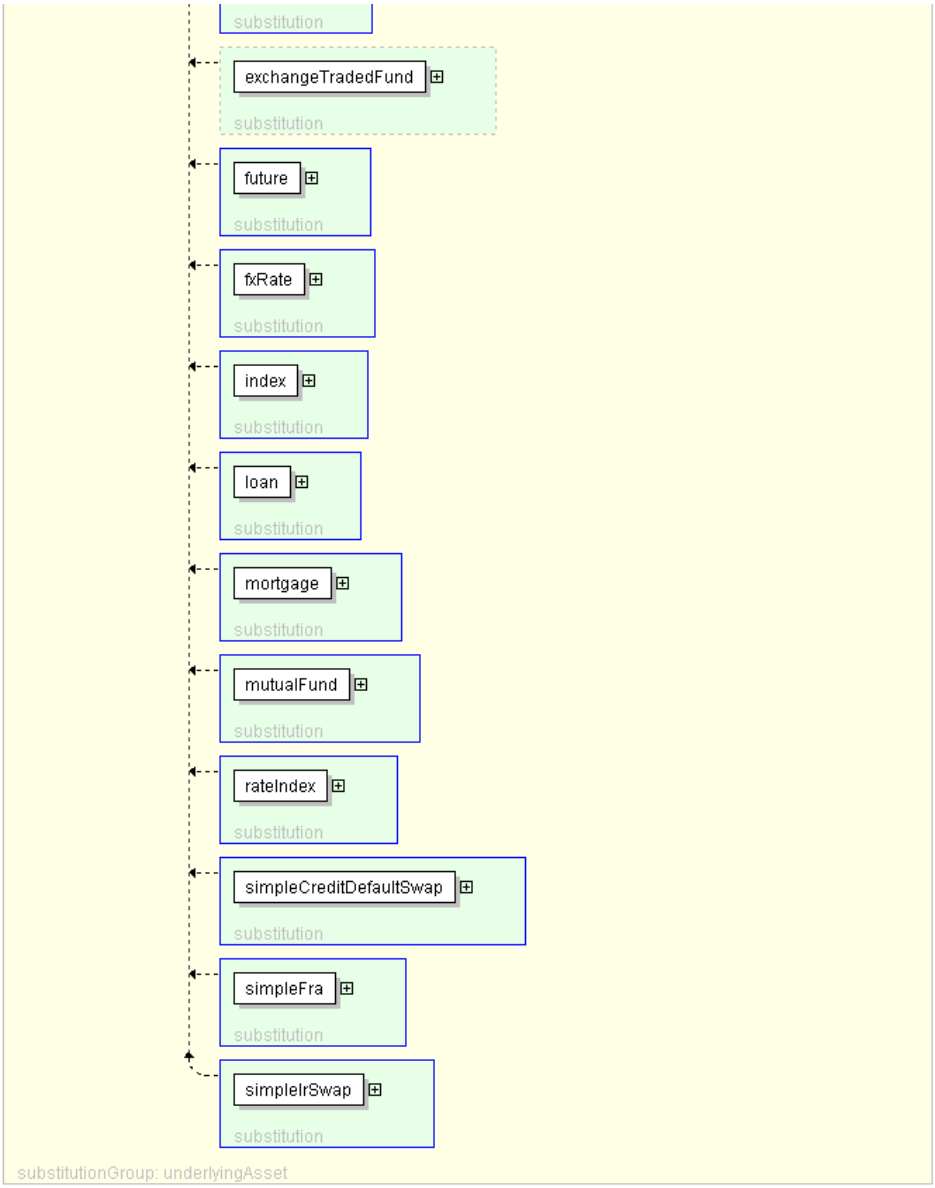
XML Instance Representation

```
<exchangeTradedFund
 id=" xsd:ID [0..1]">
  <instrumentId> InstrumentId </instrumentId> [1..*]
```

```
'Identification of the underlying asset, using public and/or private identifiers.'

<description> xsd:string </description> [0..1]
'Long name of the underlying asset.'

<currency> IdentifiedCurrency </currency> [0..1]
'Trading currency of the underlyer when transacted as a cash instrument.'

<exchangeId> ExchangeId </exchangeId> [0..1]
'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural
payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives
Definitions.'

<clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
'Identification of the clearance system associated with the transaction exchange.'

<definition> ProductReference </definition> [0..1]
'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
inconsistency between the terms of the simple product and those of the detailed definition, the values in the simple
product override those in the detailed definition.'

<relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
'A short form unique identifier for a related exchange. If the element is not present then the exchange shall be the
primary exchange on which listed futures and options on the underlying are listed. The term \"Exchange\" is assumed
to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

<optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
'A short form unique identifier for an exchange on which the reference option contract is listed. This is to address
the case where the reference exchange for the future is different than the one for the option. The options Exchange
is referenced on share options when Merger Elections are selected as Options Exchange Adjustment.'

<constituentExchangeId> ExchangeId </constituentExchangeId> [0..*]
'Identification of all the exchanges where constituents are traded. The term \"Exchange\" is assumed to have the
meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

<fundManager> xsd:string </fundManager> [0..1]
'Specifies the fund manager that is in charge of the fund.'

</exchangeTradedFund>
```

Diagram





Schema Component Representation

```
<xsd:element name="exchangeTradedFund" type=" ExchangeTradedFund " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

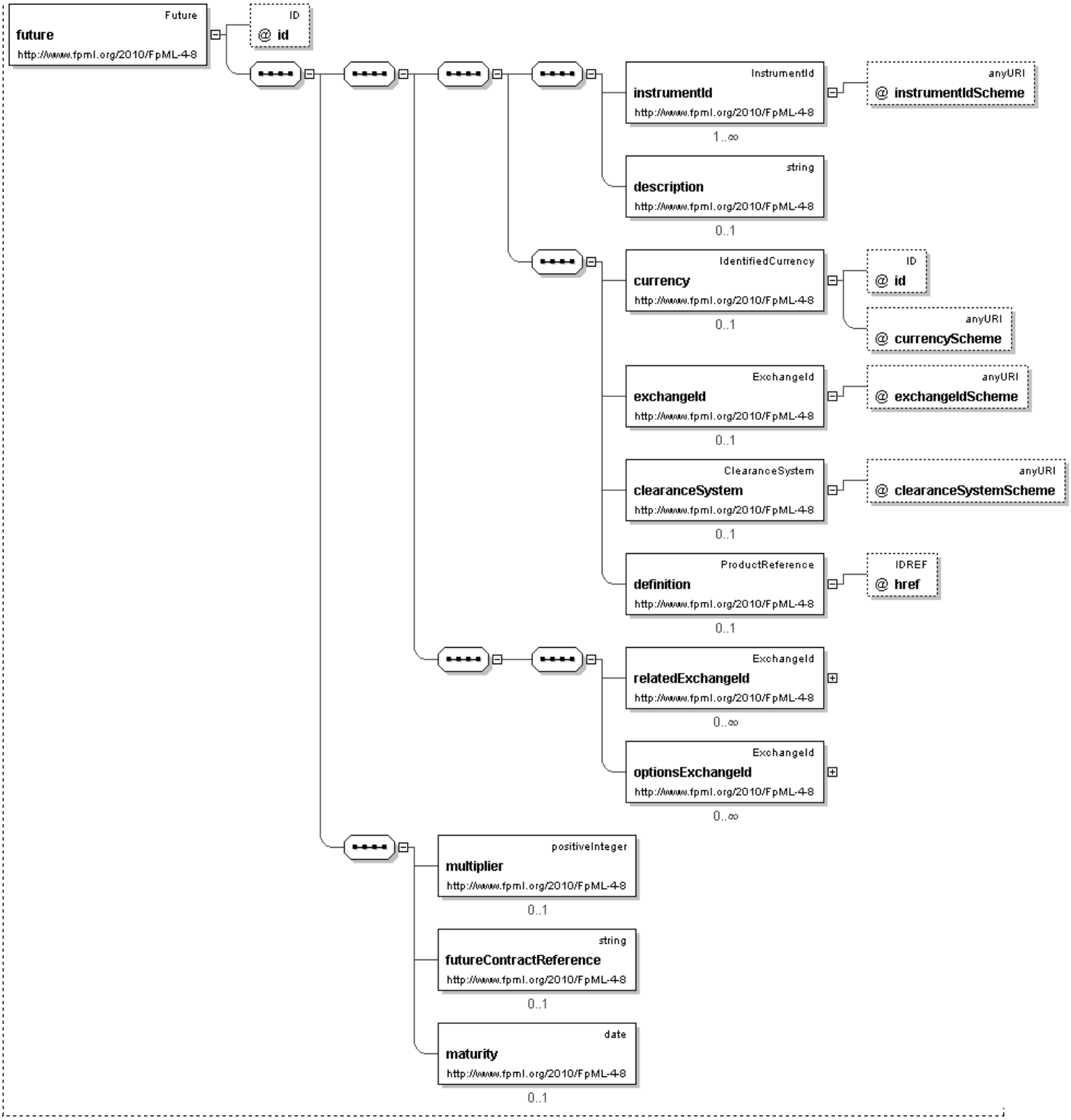
Element: future

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	future
Type	Future
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a listed future contract.

Logical Diagram

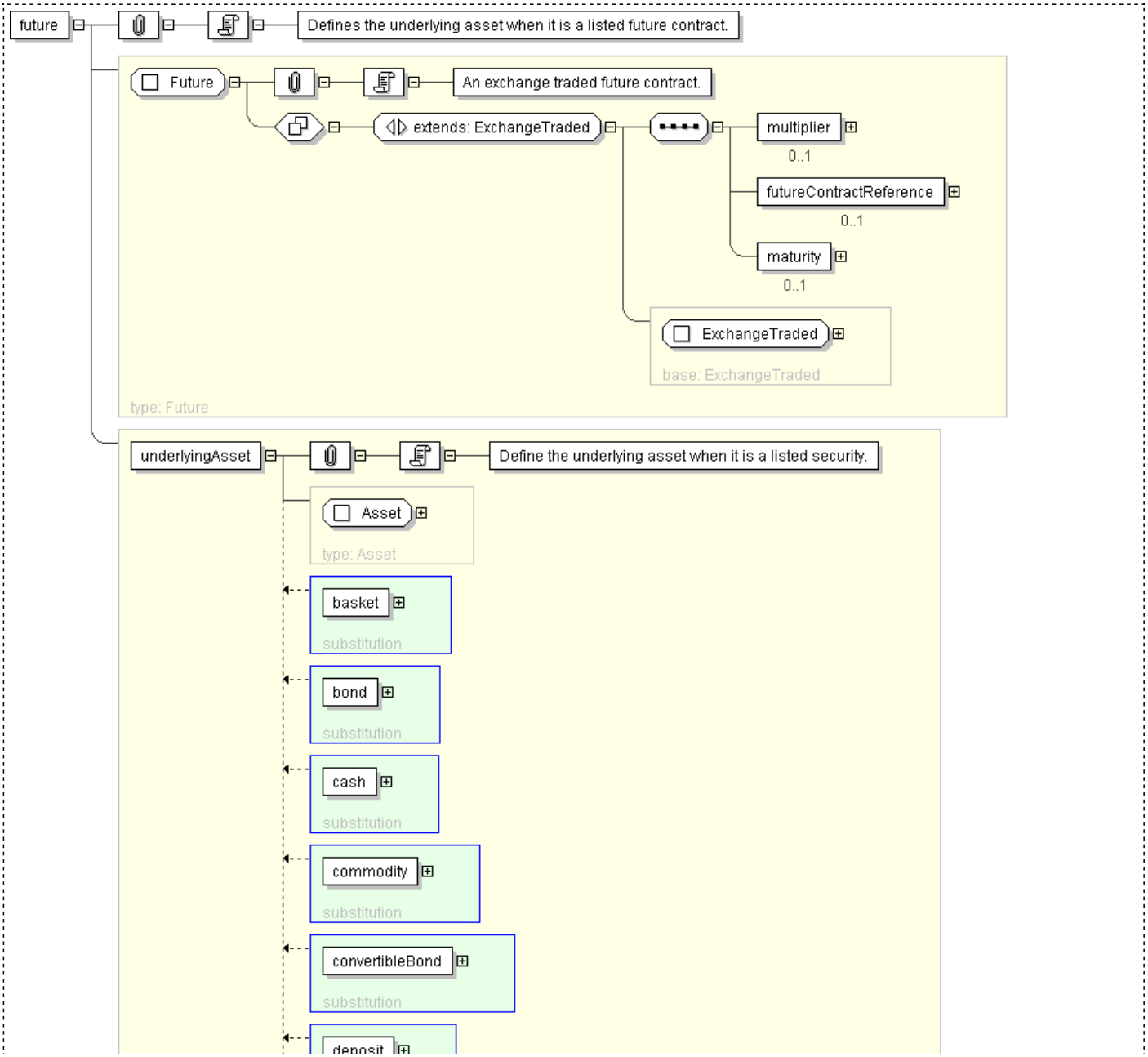


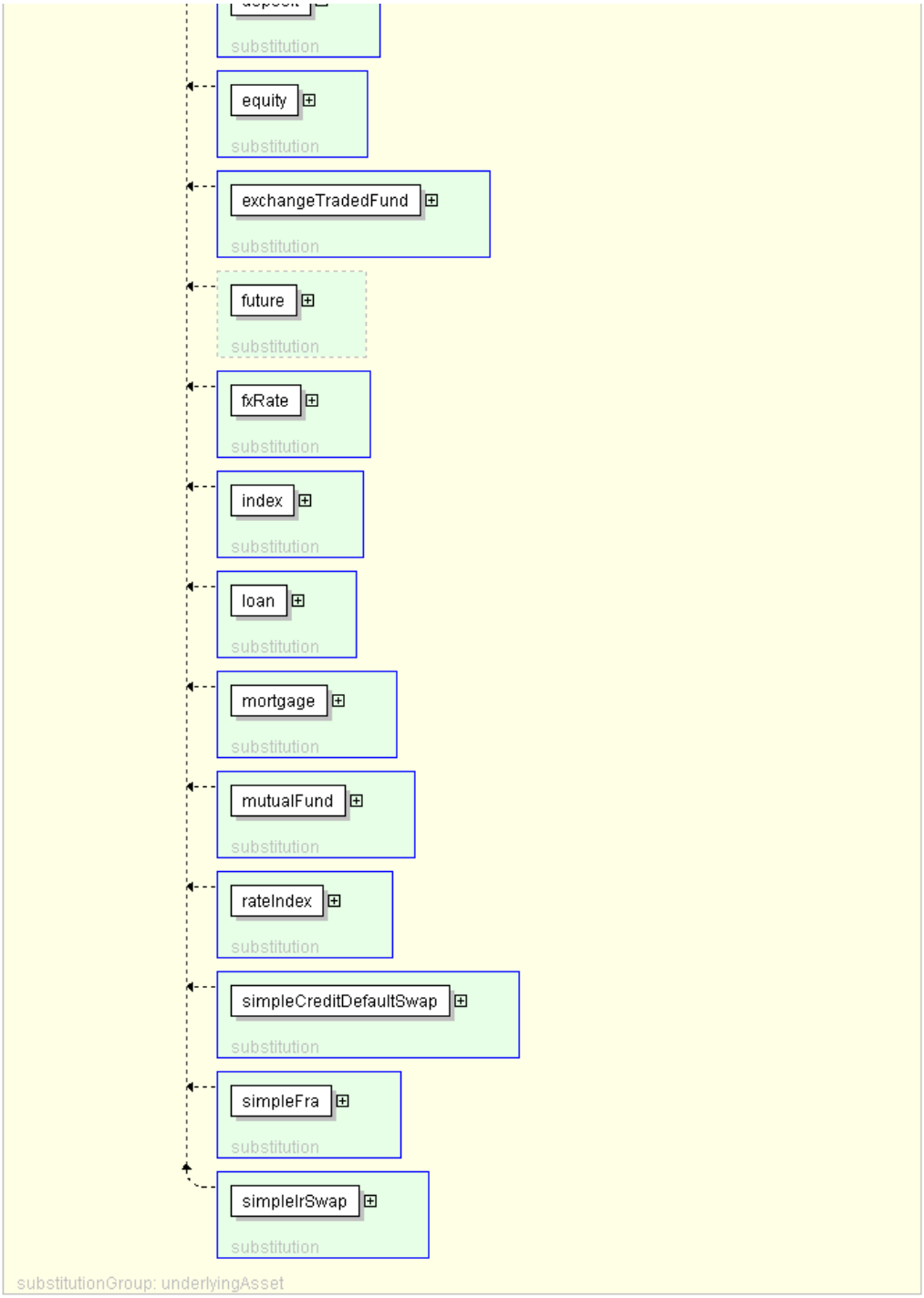
XML Instance Representation

```
<future
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a
   contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002
   Equity Derivatives Definitions.'
```

```
<clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
'Identification of the clearance system associated with the transaction exchange.'

<definition> ProductReference </definition> [0..1]
'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
inconsistency between the terms of the simple product and those of the detailed definition, the values in
the simple product override those in the detailed definition.'

<relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
'A short form unique identifier for a related exchange. If the element is not present then the exchange
shall be the primary exchange on which listed futures and options on the underlying are listed. The term
\"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

<optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
'A short form unique identifier for an exchange on which the reference option contract is listed. This is
to address the case where the reference exchange for the future is different than the one for the option.
The options Exchange is referenced on share options when Merger Elections are selected as Options Exchange
Adjustment.'

<multiplier> xsd:positiveInteger </multiplier> [0..1]
'Specifies the contract multiplier that can be associated with the number of units.'

<futureContractReference> xsd:string </futureContractReference> [0..1]
'Specifies the future contract that can be referenced, besides the equity or index reference defined as
part of the UnderlyerAsset type.'

<maturity> xsd:date </maturity> [0..1]
'The date when the future contract expires.'

</future>
```

Diagram





Schema Component Representation

```
<xsd:element name="future" type=" Future " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

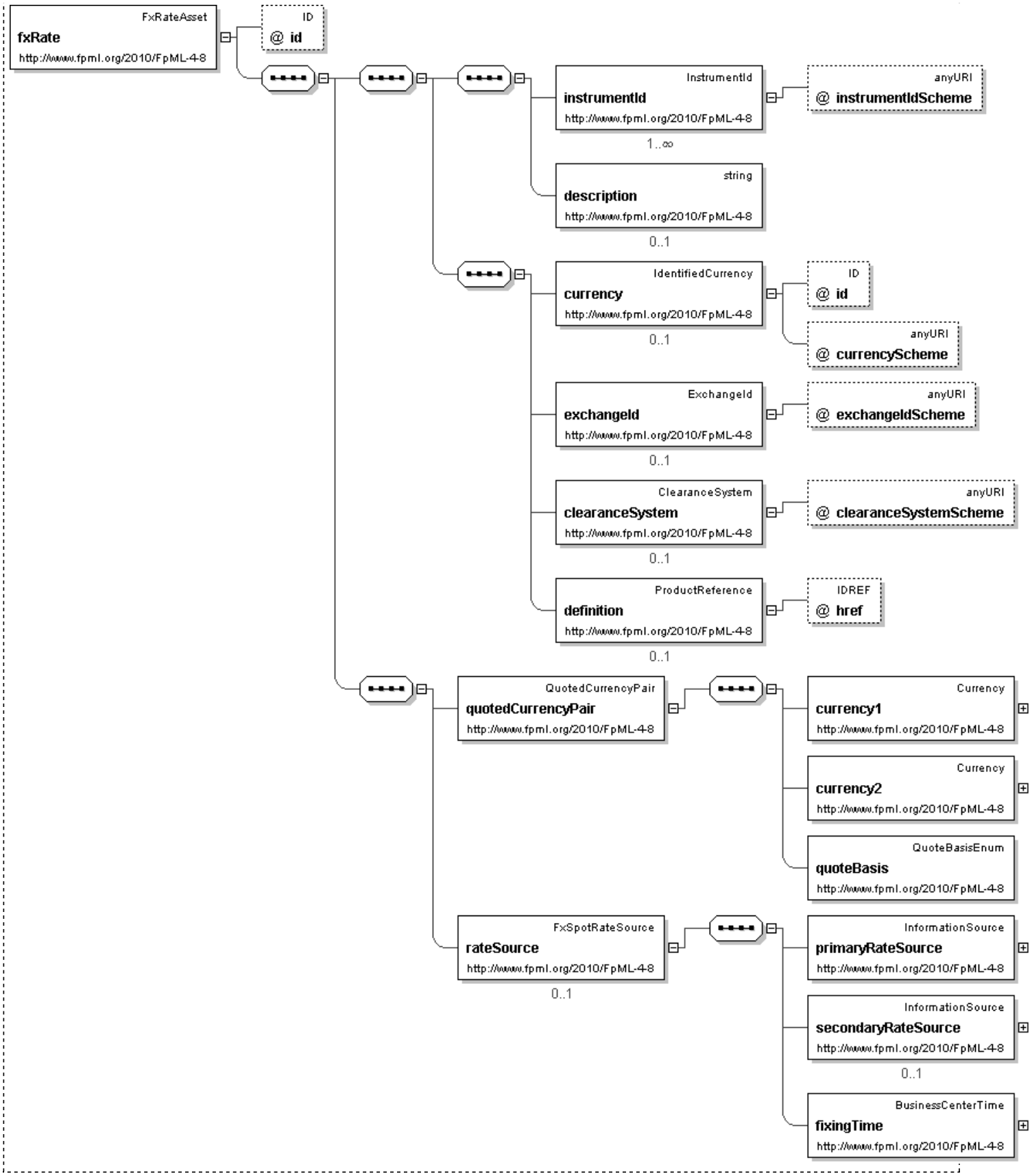
Element: fxRate

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	fxRate
Type	FxRateAsset
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset type that is an FX rate. Used for specifying FX rates in the pricing and risk model.

Logical Diagram

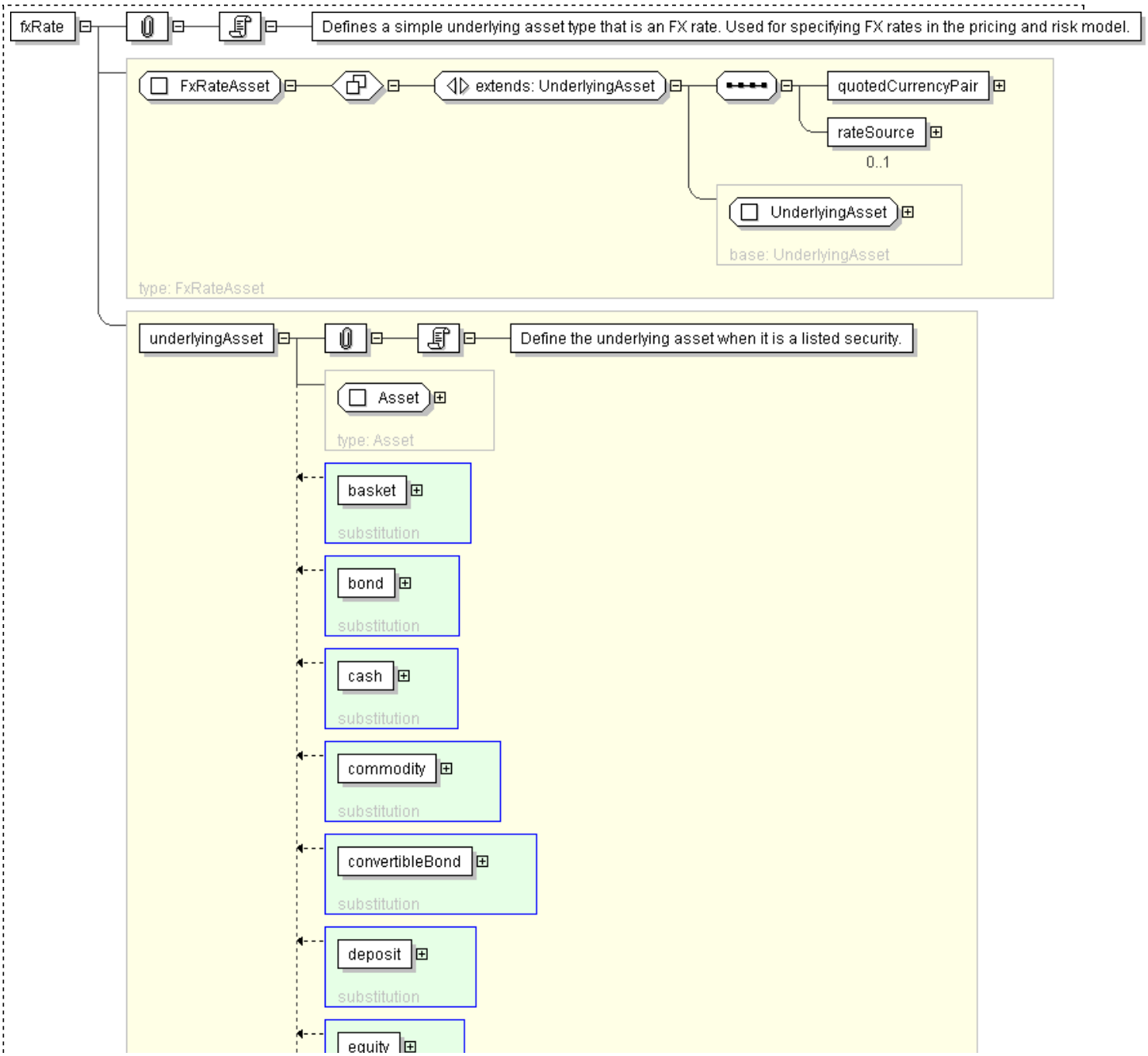


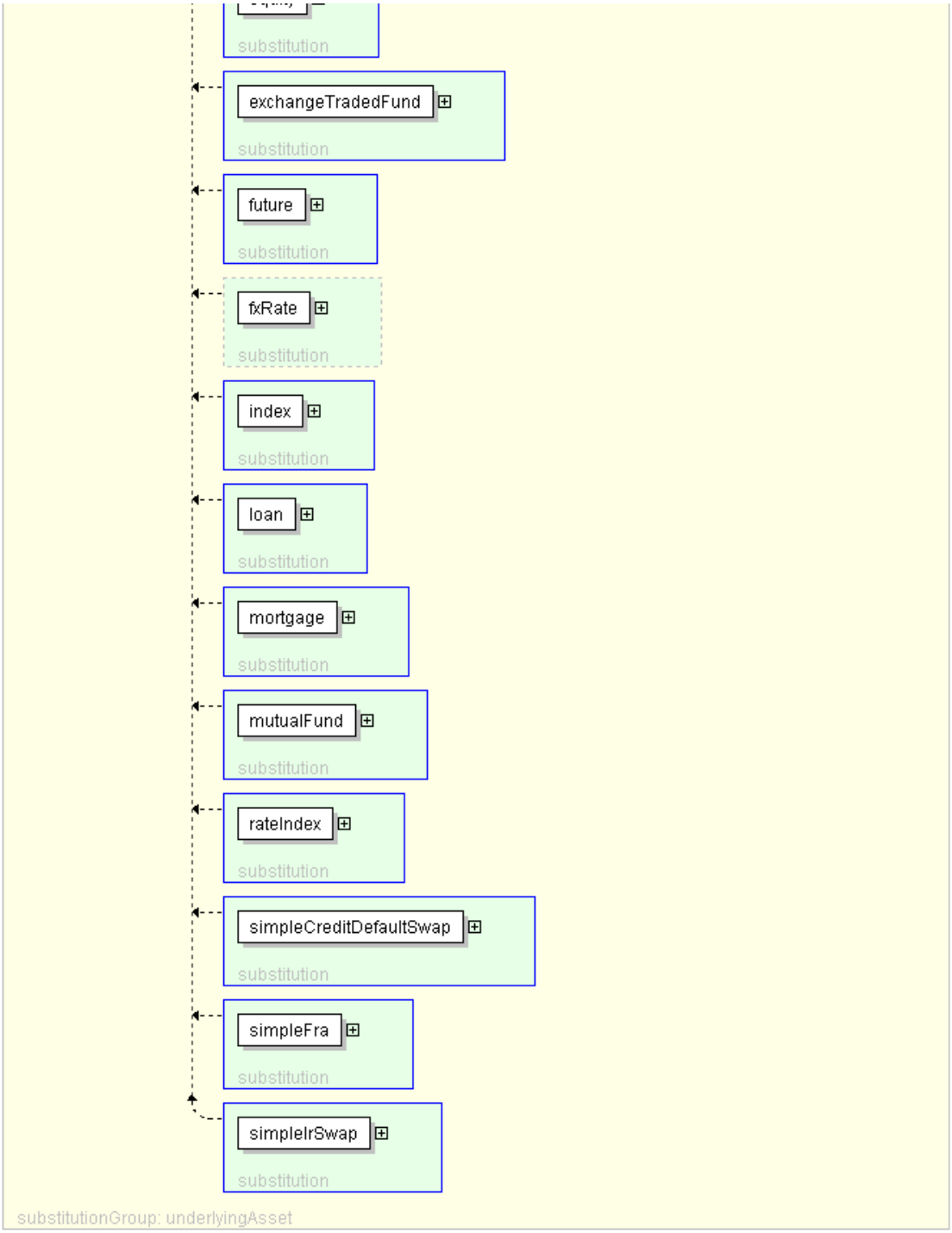
XML Instance Representation

```
<fxRate
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'
```

```
    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating
    a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
    2002 Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail.
    In case of inconsistency between the terms of the simple product and those of the detailed
    definition, the values in the simple product override those in the detailed definition.'

    <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
    'Defines the two currencies for an FX trade and the quotation relationship between the two
    currencies.'

    <rateSource> FxSpotRateSource </rateSource> [0..1]
    'Defines the source of the FX rate.'

</fxRate>
```

Diagram





Schema Component Representation

```
<xsd:element name="fxRate" type=" FxRateAsset " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

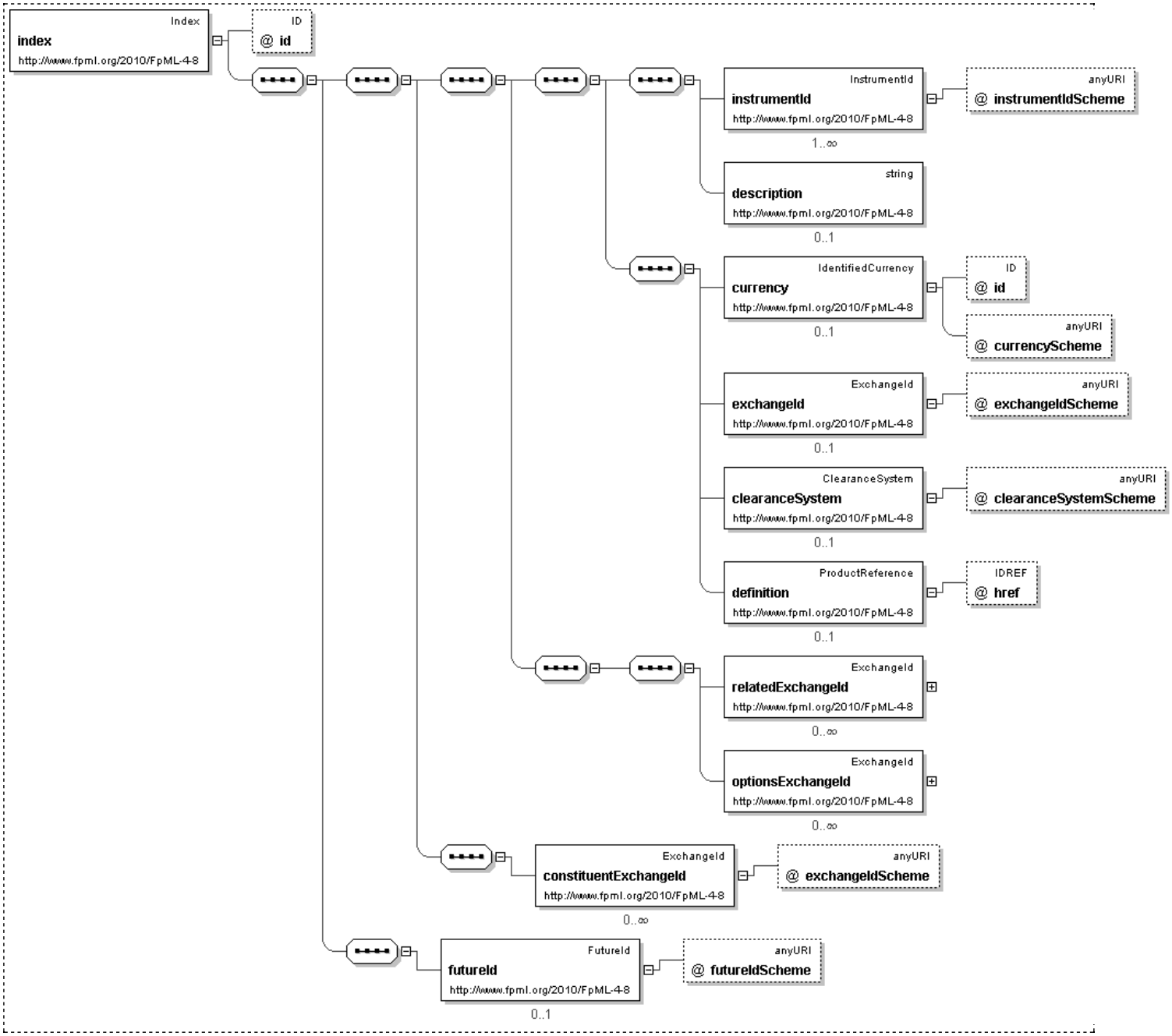
Element: index

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	index
Type	Index
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a financial index.

Logical Diagram

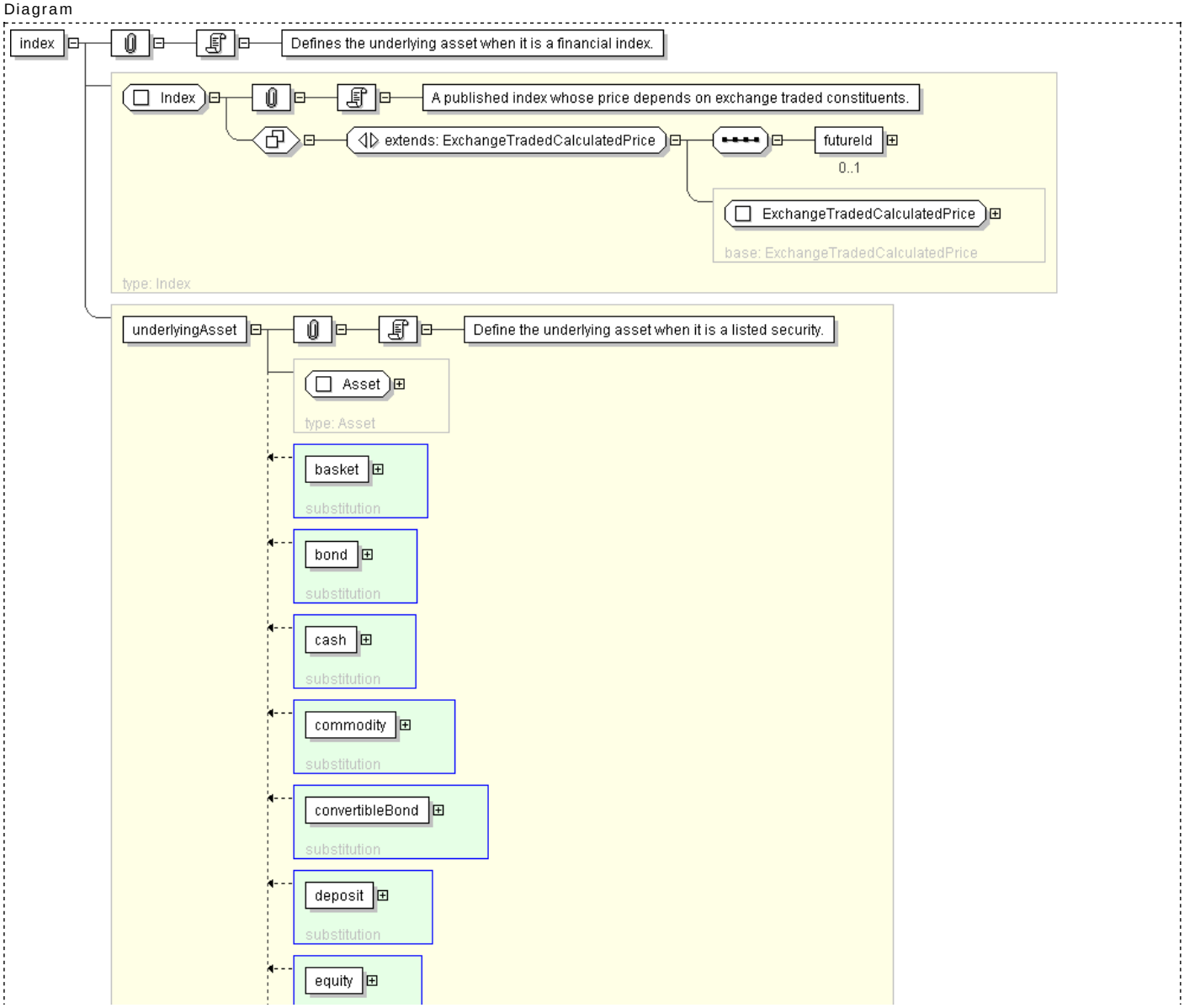


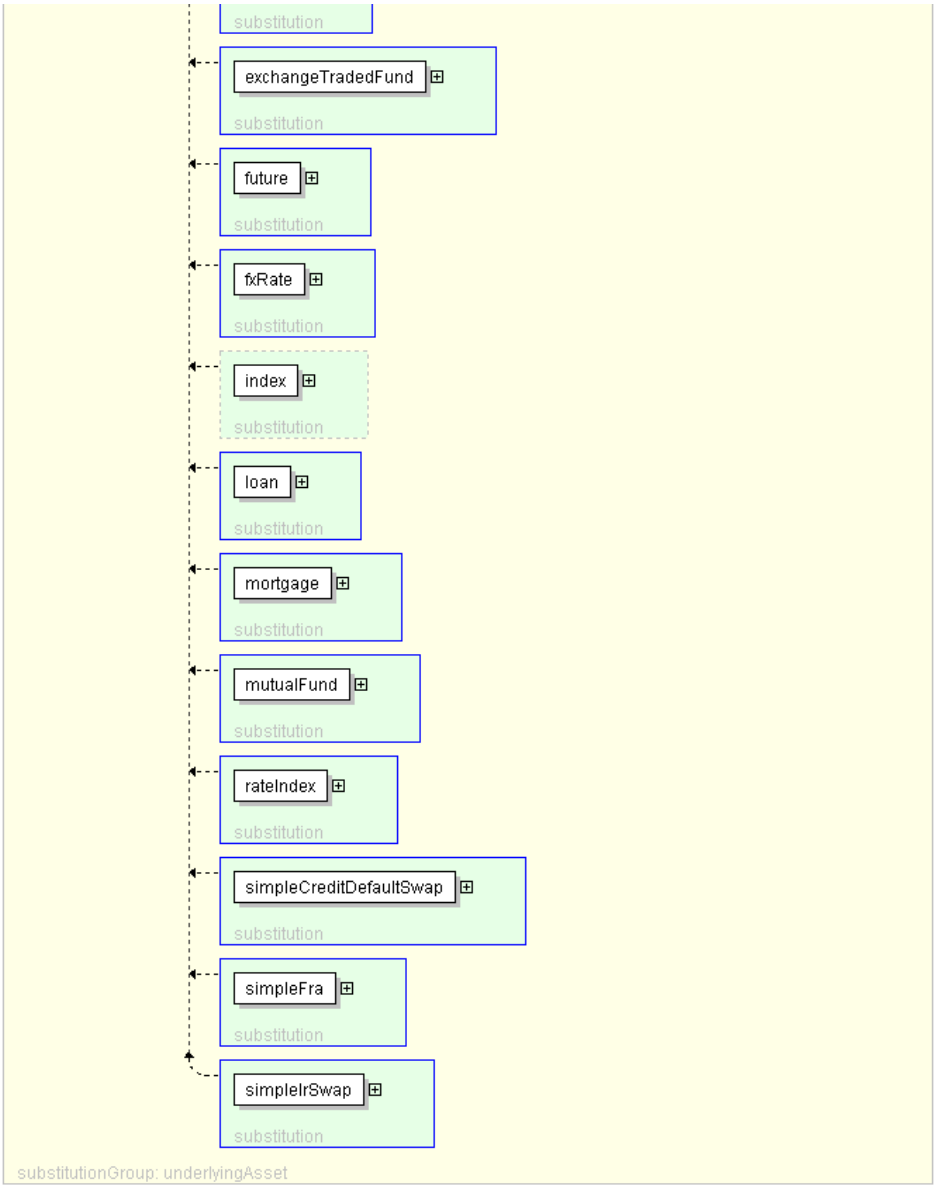
XML Instance Representation

```
<index
 id=" xsd:ID [0..1]">
  <instrumentId> InstrumentId </instrumentId> [1..*]
```

```
    'Identification of the underlying asset, using public and/or private identifiers.'

    <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural
    payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
    inconsistency between the terms of the simple product and those of the detailed definition, the values in the simple
    product override those in the detailed definition.'

    <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not present then the exchange shall be the
    primary exchange on which listed futures and options on the underlying are listed. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

    <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option contract is listed. This is to address
    the case where the reference exchange for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options Exchange Adjustment.'

    <constituentExchangeId> ExchangeId </constituentExchangeId> [0..*]
    'Identification of all the exchanges where constituents are traded. The term \"Exchange\" is assumed to have the
    meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

    <futureId> FutureId </futureId> [0..1]
    'A short form unique identifier for the reference future contract in the case of an index underlyer.'

</index>
```

Diagram





Schema Component Representation

```
<xsd:element name="index" type=" Index " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

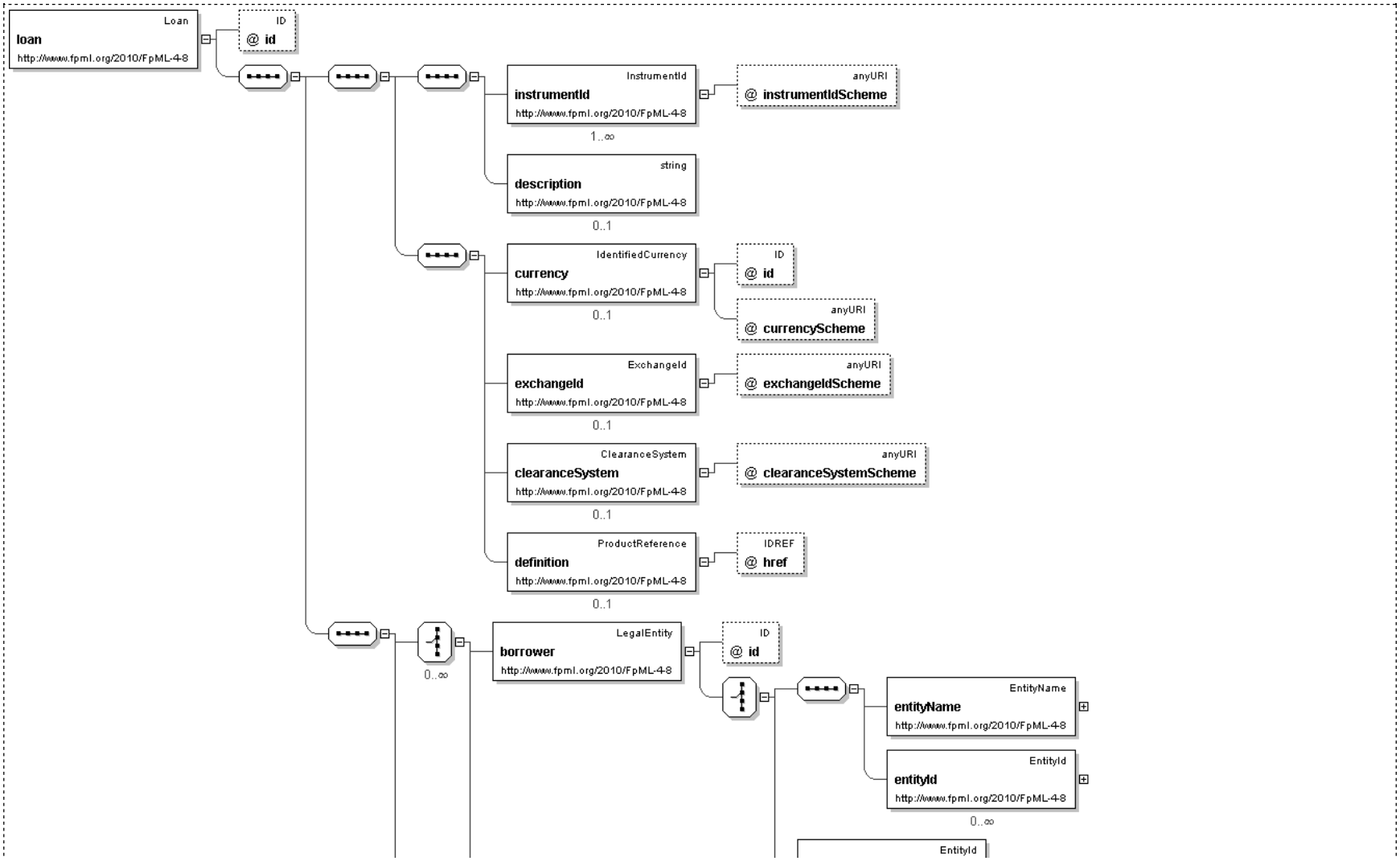
Element: loan

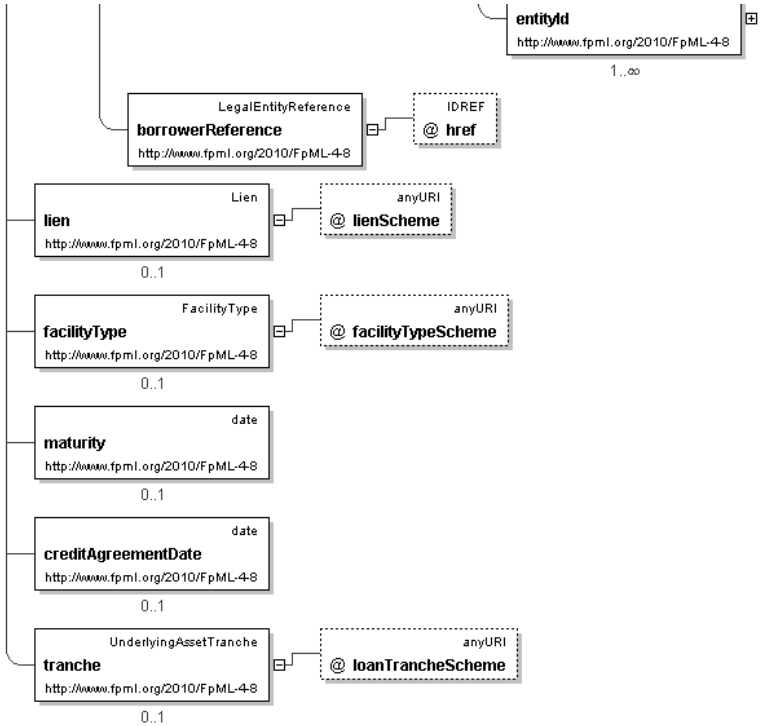
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	loan
Type	Loan
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is a loan.

Logical Diagram



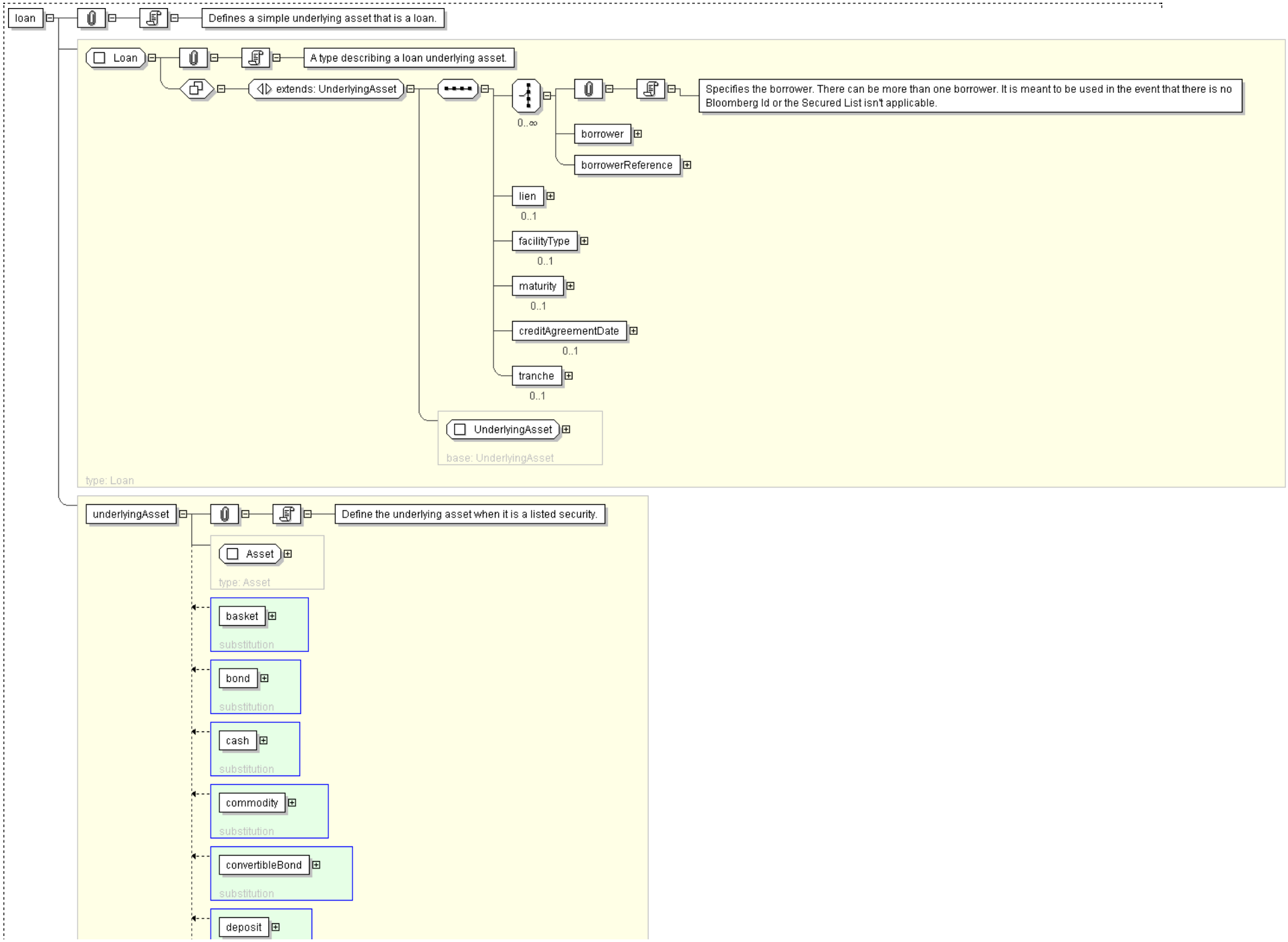


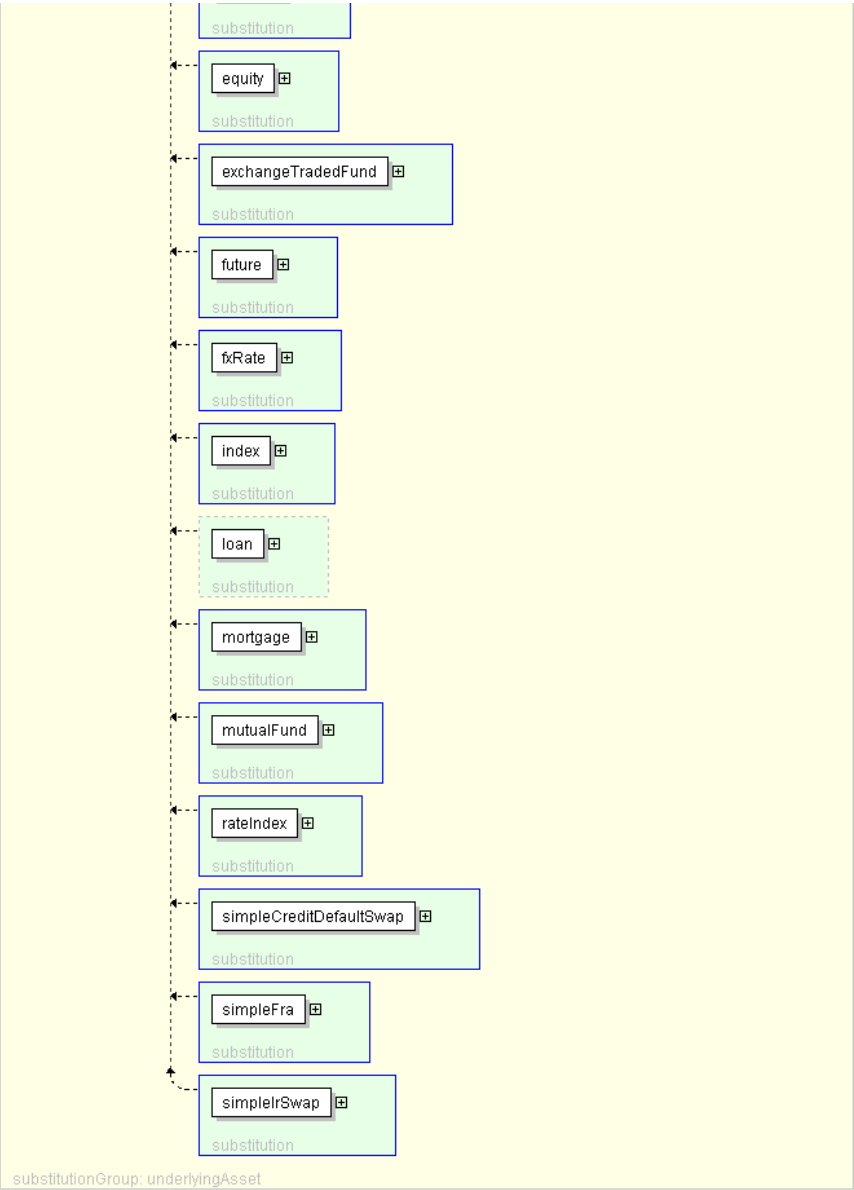
XML Instance Representation

```
<loan
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural payoff. The term \"Exchange\" is assumed to have
   the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail. In case of inconsistency between the terms of the simple product
   and those of the detailed definition, the values in the simple product override those in the detailed definition.'

   Start Choice [0..*]
 'Specifies the borrower. There can be more than one borrower. It is meant to be used in the event that there is no Bloomberg Id or the Secured List isn\'t
 applicable.'


      <borrower> LegalEntity </borrower> [1]
      <borrowerReference> LegalEntityReference </borrowerReference> [1]
   End Choice
   <lien> Lien </lien> [0..1]
   'Specifies the seniority level of the lien.'

   <facilityType> FacilityType </facilityType> [0..1]
   'The type of loan facility (letter of credit, revolving, ...).'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of the loan becomes due and payable.'

   <creditAgreementDate> xsd:date </creditAgreementDate> [0..1]
   'The credit agreement date is the closing date (the date where the agreement has been signed) for the loans in the credit agreement. Funding of the facilities
   occurs on (or sometimes a little after) the Credit Agreement date. This underlyer attribute is used to help identify which of the company\'s outstanding loans
   are being referenced by knowing to which credit agreement it belongs. ISDA Standards Terms Supplement term: Date of Original Credit Agreement.'
```

```
  <tranche> UnderlyingAssetTranche </tranche> [0..1]
  'The loan tranche that is subject to the derivative transaction. It will typically be referenced as the Bloomberg tranche number. ISDA Standards Terms Supplement term: Bloomberg Tranche Number.'

</loan>
```

Diagram





Schema Component Representation

```
<xsd:element name="loan" type=" Loan " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

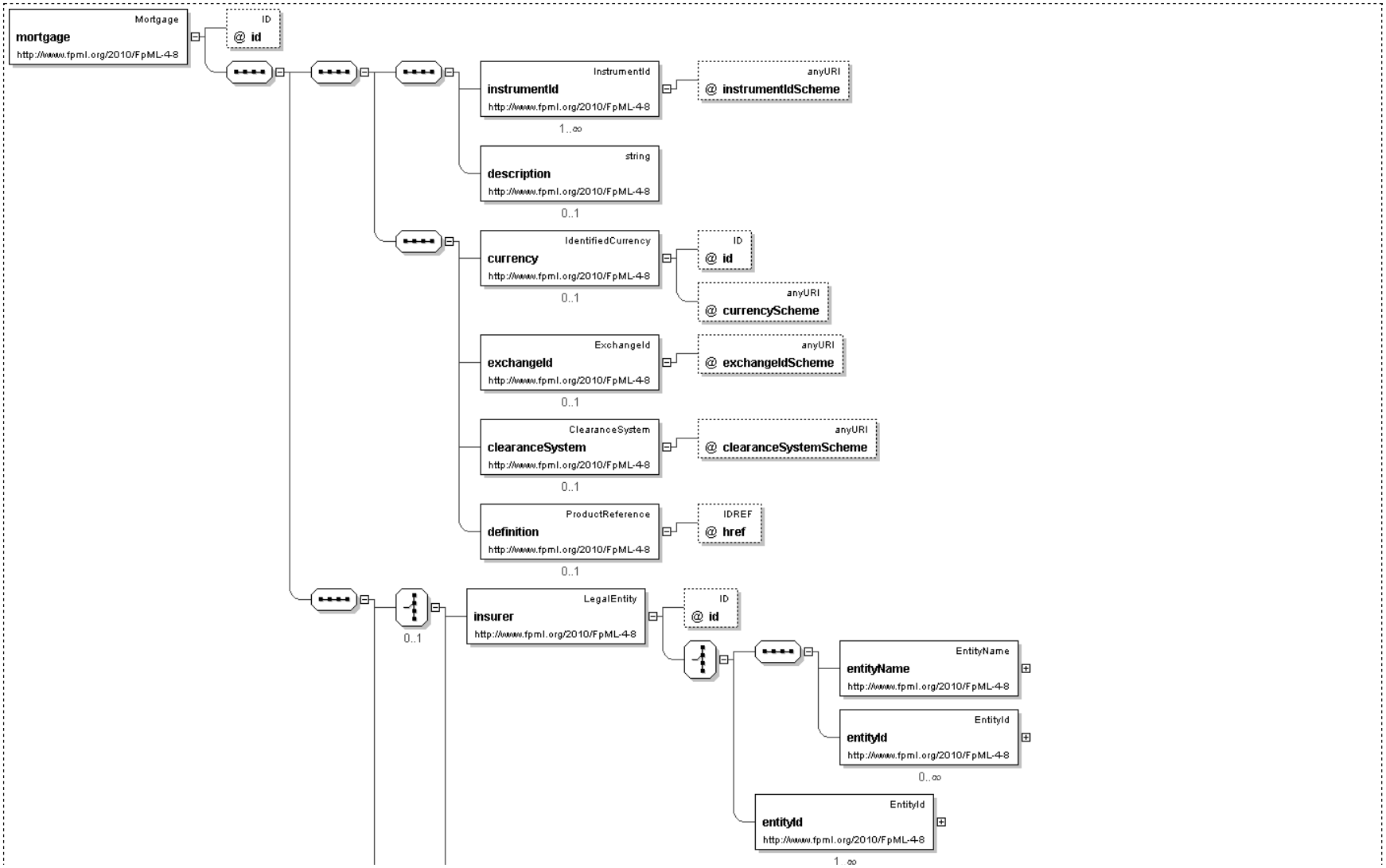
Element: mortgage

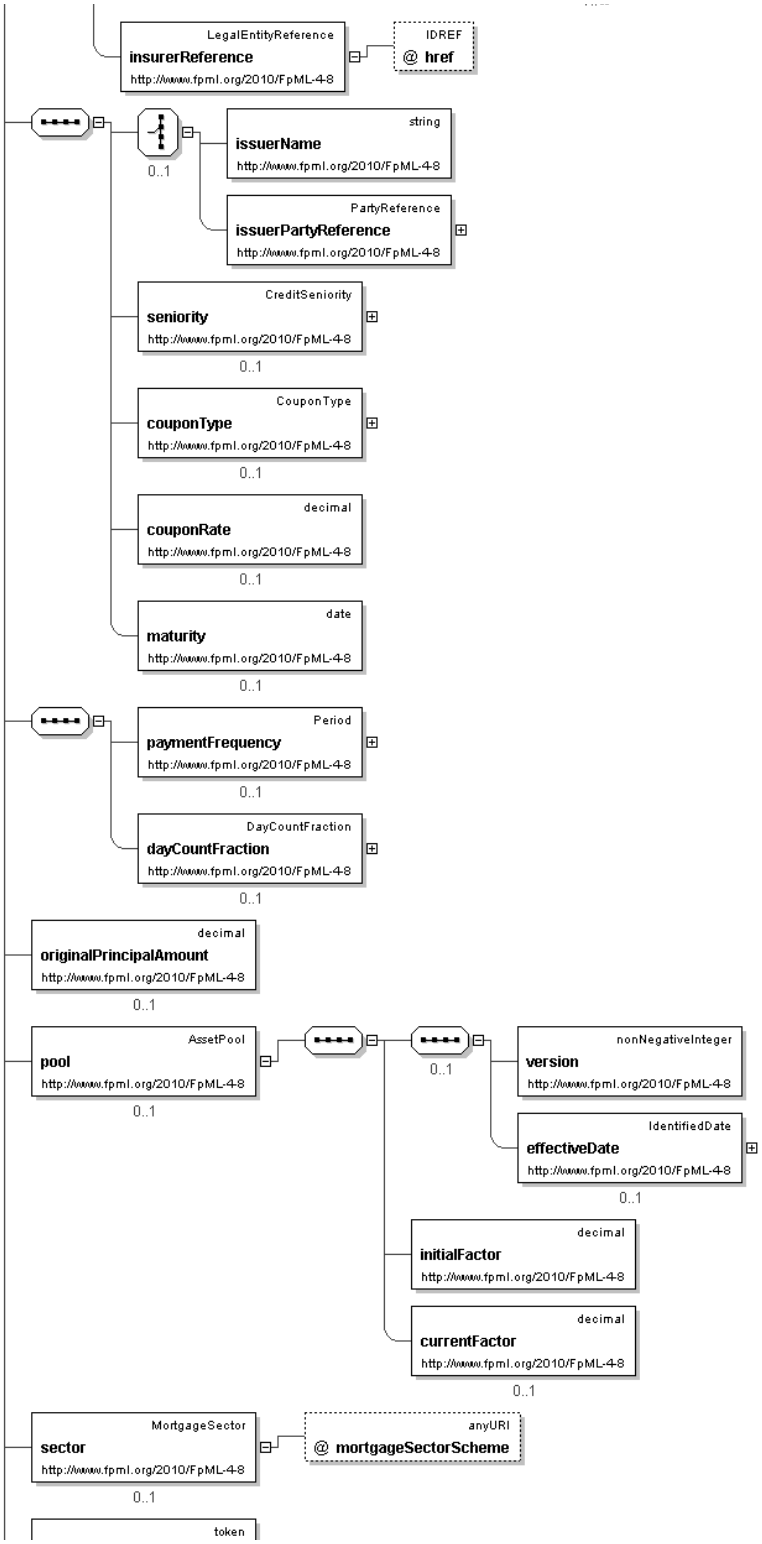
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	mortgage
Type	Mortgage
Nillable	no
Abstract	no
Documentation	Defines an underlying asset that is a mortgage.

Logical Diagram

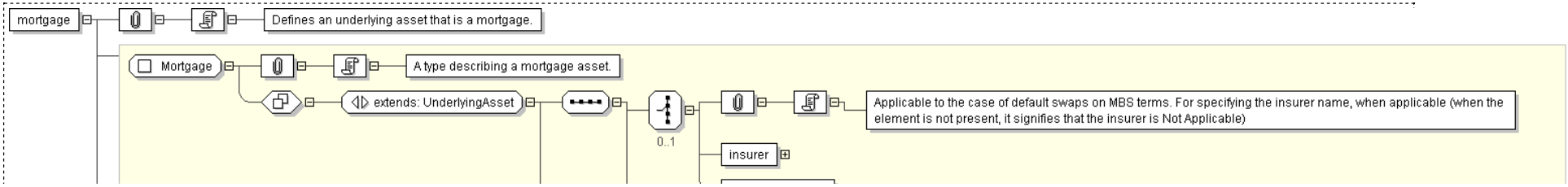


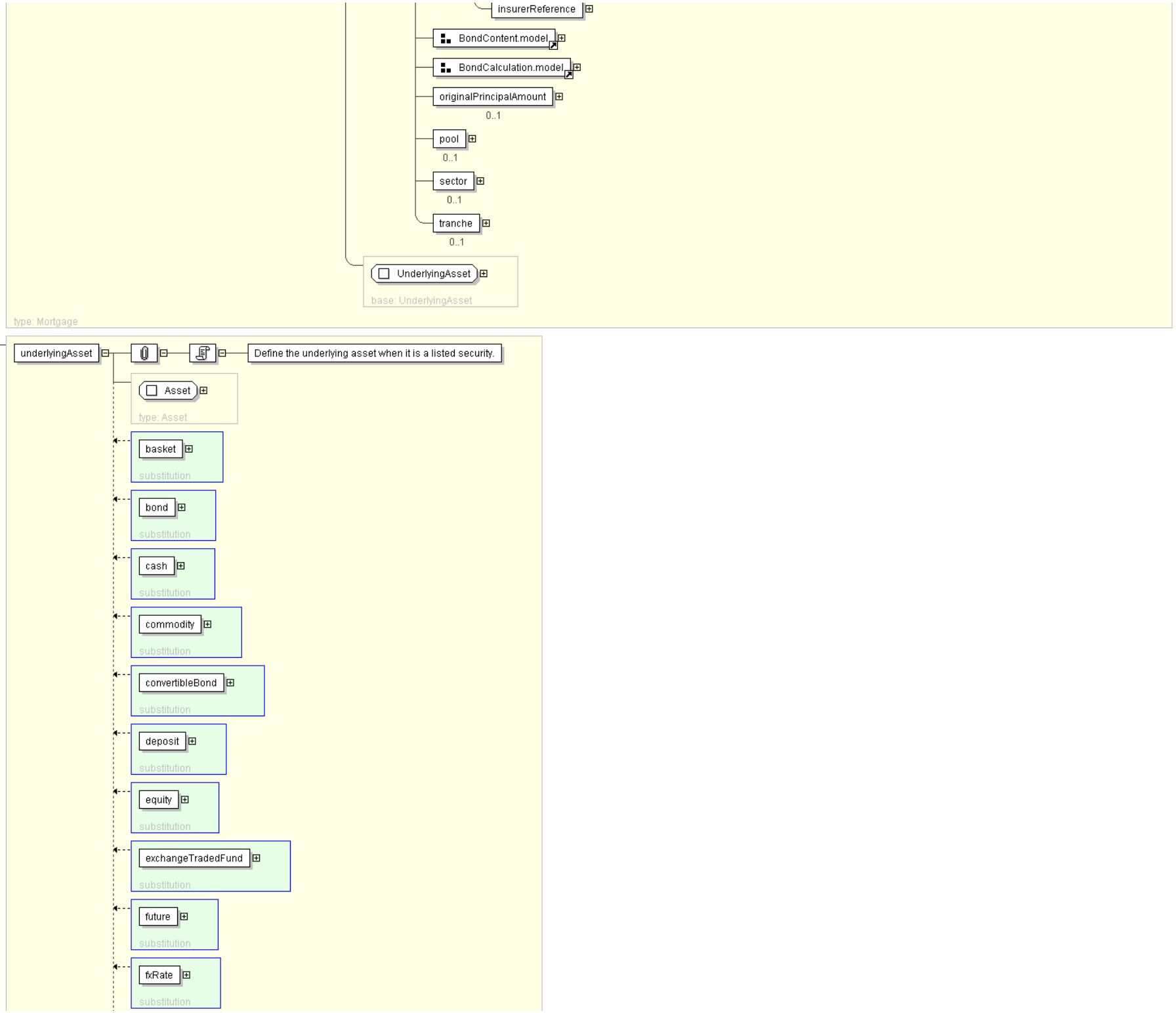

LegalEntityReference
insurerReference
http://www.fpml.org/2010/FpML-4-8
IDREF
@ href
0..1
string
issuerName
http://www.fpml.org/2010/FpML-4-8
PartyReference
issuerPartyReference
http://www.fpml.org/2010/FpML-4-8
CreditSeniority
seniority
http://www.fpml.org/2010/FpML-4-8
0..1
CouponType
couponType
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
couponRate
http://www.fpml.org/2010/FpML-4-8
0..1
date
maturity
http://www.fpml.org/2010/FpML-4-8
0..1
Period
paymentFrequency
http://www.fpml.org/2010/FpML-4-8
0..1
DayCountFraction
dayCountFraction
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
originalPrincipalAmount
http://www.fpml.org/2010/FpML-4-8
0..1
AssetPool
pool
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
nonNegativeInteger
version
http://www.fpml.org/2010/FpML-4-8
IdentifiedDate
effectiveDate
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
initialFactor
http://www.fpml.org/2010/FpML-4-8
decimal
currentFactor
http://www.fpml.org/2010/FpML-4-8
0..1
MortgageSector
sector
http://www.fpml.org/2010/FpML-4-8
0..1
anyURI
@ mortgageSectorScheme
token

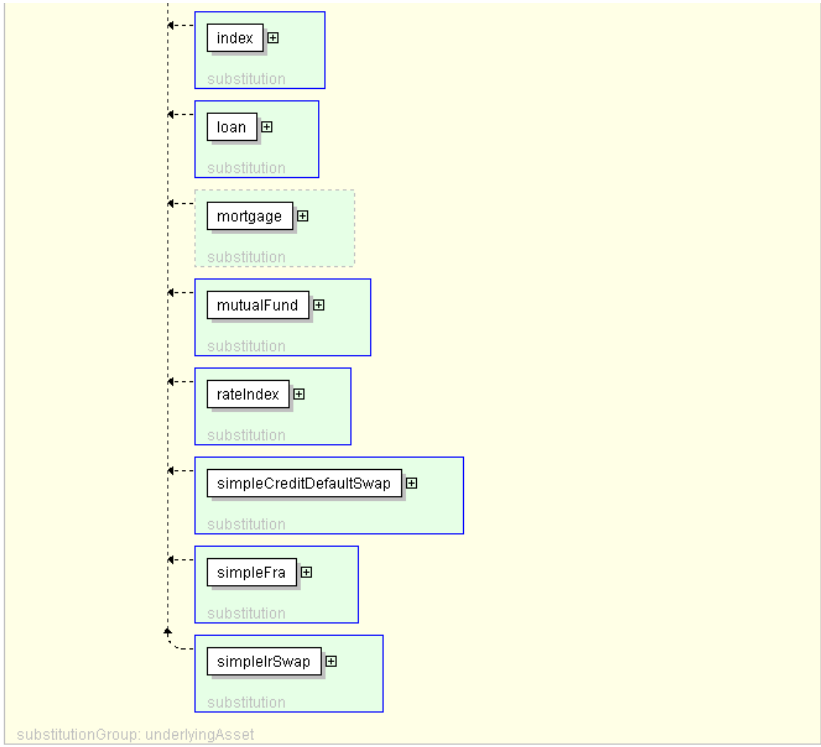
XML Instance Representation

```
<mortgage
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural payoff. The term \"Exchange\" is assumed to have the
   meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail. In case of inconsistency between the terms of the simple product and
   those of the detailed definition, the values in the simple product override those in the detailed definition.'

   Start Choice [0..1]
'Applicable to the case of default swaps on MBS terms. For specifying the insurer name, when applicable (when the element is not present, it signifies that the insurer is
Not Applicable)'


      <insurer> LegalEntity </insurer> [1]
      <insurerReference> LegalEntityReference </insurerReference> [1]
   End Choice
   Start Choice [0..1]
'Specifies the issuer name of a fixed income security or convertible bond. This name can either be explicitly stated, or specified as an href into another element of the
document, such as the obligor'


      <issuerName> xsd:string </issuerName> [1]
      <issuerPartyReference> PartyReference </issuerPartyReference> [1]
   End Choice
   <seniority> CreditSeniority </seniority> [0..1]
   'The repayment precedence of a debt instrument.'

   <couponType> CouponType </couponType> [0..1]
   'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-coupon.'

   <couponRate> xsd:decimal </couponRate> [0..1]
   'Specifies the coupon rate (expressed in percentage) of a fixed income security or convertible bond.'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of a security becomes due and payable.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
   'Specifies the frequency at which the bond pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
   'The day count basis for the bond.'

   <originalPrincipalAmount> xsd:decimal </originalPrincipalAmount> [0..1]
   'The initial issued amount of the mortgage obligation.'

   <pool> AssetPool </pool> [0..1]
   'The morgage pool that is underneath the mortgage obligation.'

   <sector> MortgageSector </sector> [0..1]
   'The sector classification of the mortgage obligation.'

   <tranche> xsd:token </tranche> [0..1]
   'The mortgage obligation tranche that is subject to the derivative transaction.'

</mortgage>
```

Diagram




insurerReference
BondContent.model
BondCalculation.model
originalPrincipalAmount
0..1
pool
0..1
sector
0..1
tranche
0..1
UnderlyingAsset
base: UnderlyingAsset
type: Mortgage
underlyingAsset
Define the underlying asset when it is a listed security.
Asset
type: Asset
basket
substitution
bond
substitution
cash
substitution
commodity
substitution
convertibleBond
substitution
deposit
substitution
equity
substitution
exchangeTradedFund
substitution
future
substitution
fxRate
substitution



Schema Component Representation

```
<xsd:element name="mortgage" type=" Mortgage " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

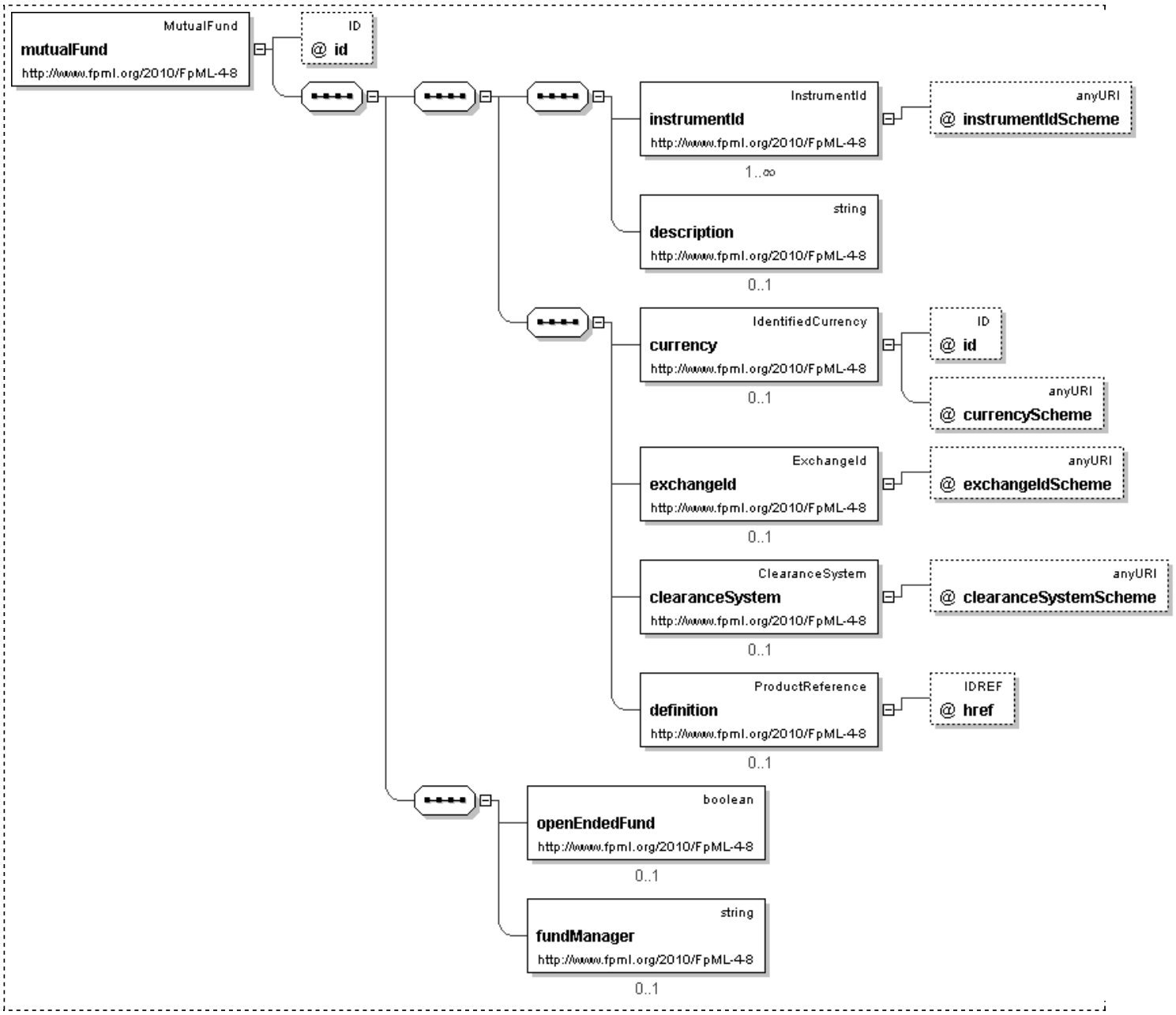
Element: mutualFund

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	mutualFund
Type	MutualFund
Nillable	no
Abstract	no
Documentation	Defines the underlying asset when it is a mutual fund.

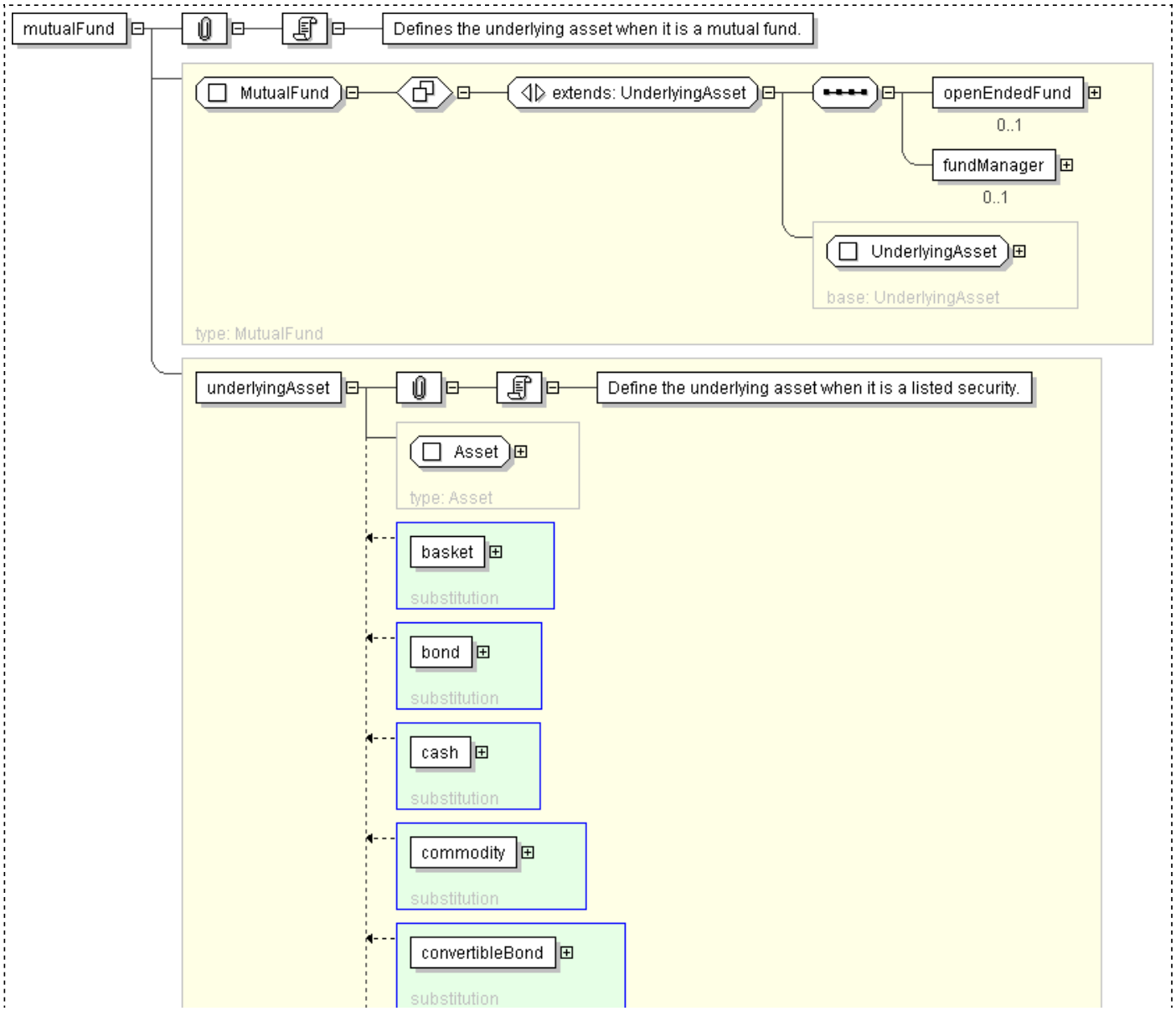
Logical Diagram

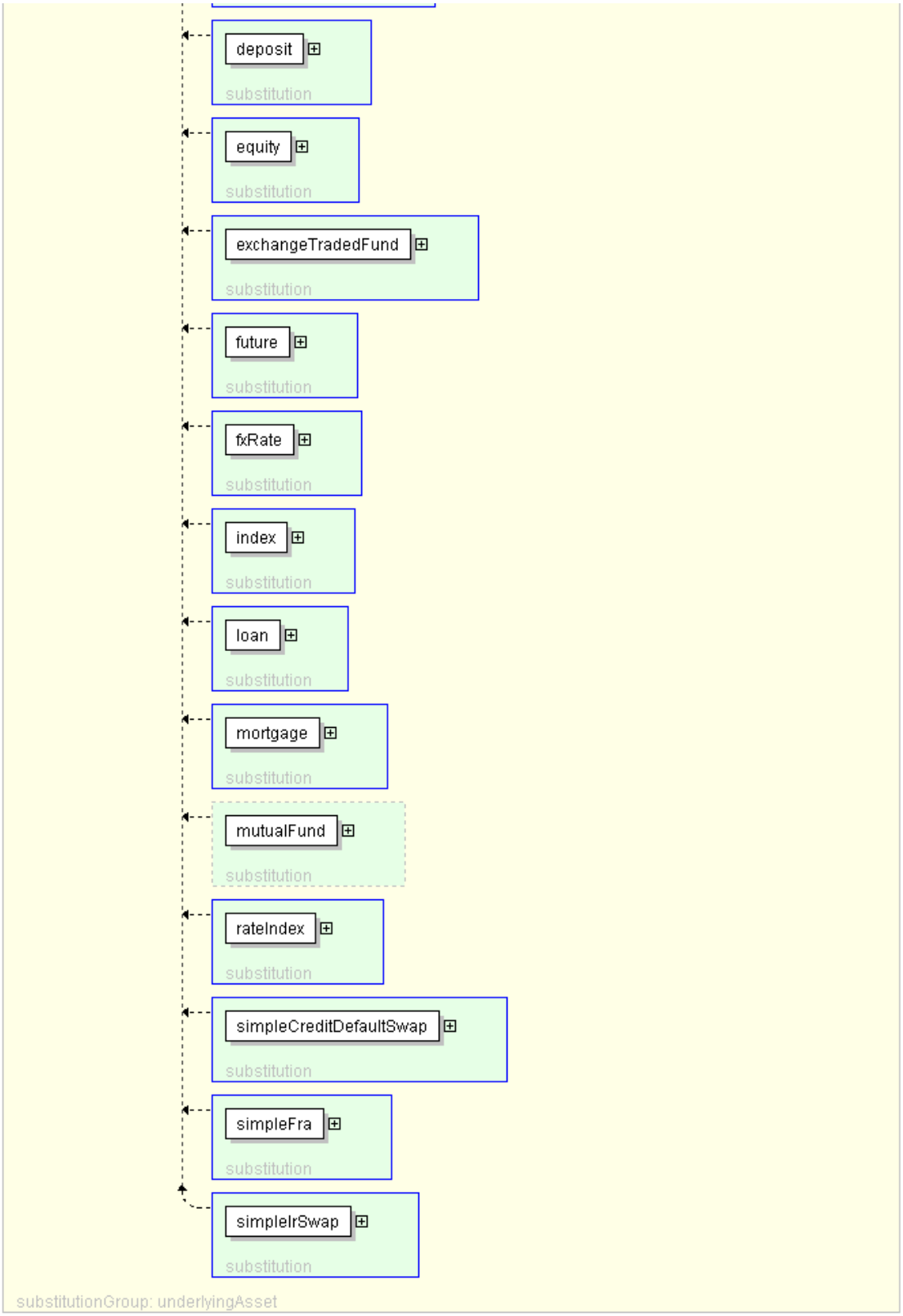


XML Instance Representation

```
<mutualFund
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating
   a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
   2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail.
   In case of inconsistency between the terms of the simple product and those of the detailed
   definition, the values in the simple product override those in the detailed definition.'

   <openEndedFund> xsd:boolean </openEndedFund> [0..1]
   'Boolean indicator to specify whether the mutual fund is an open-ended mutual fund.'

   <fundManager> xsd:string </fundManager> [0..1]
   'Specifies the fund manager that is in charge of the fund.'

</mutualFund>
```

Diagram





Schema Component Representation

```
<xsd:element name="mutualFund" type=" MutualFund " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

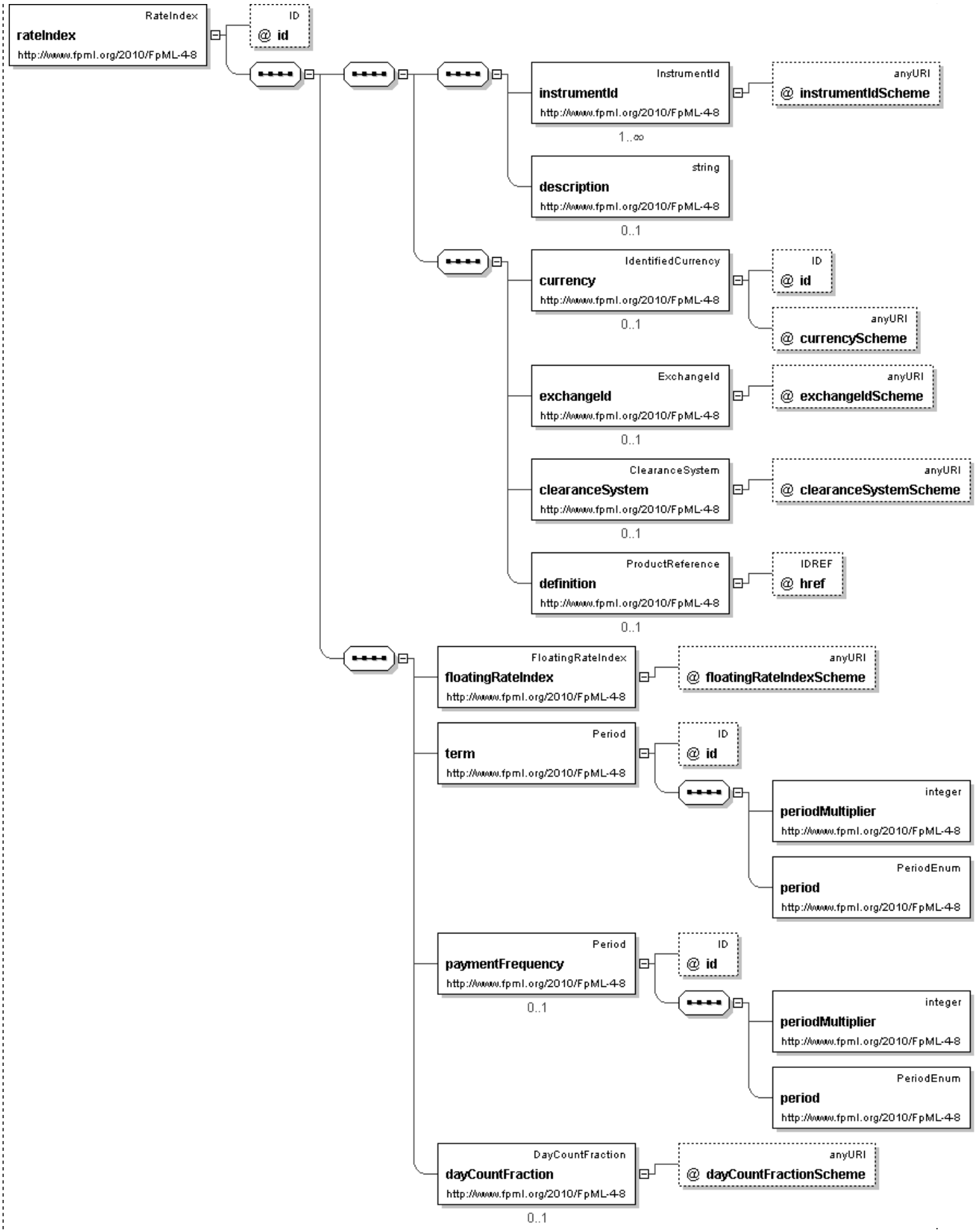
Element: rateIndex

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	rateIndex
Type	RateIndex
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is an interest rate index. Used for specifying benchmark assets in the market environment in the pricing and risk model.

Logical Diagram

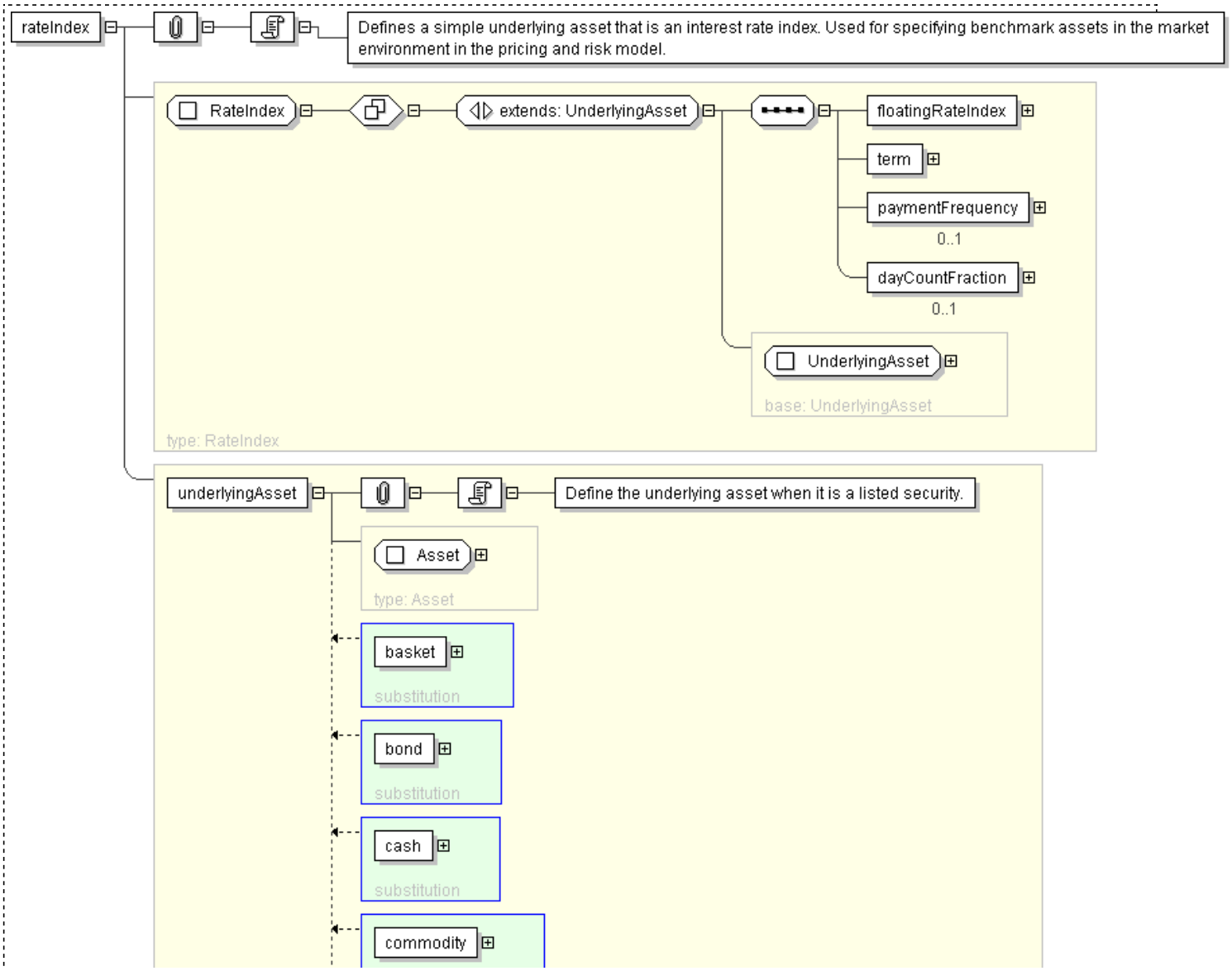


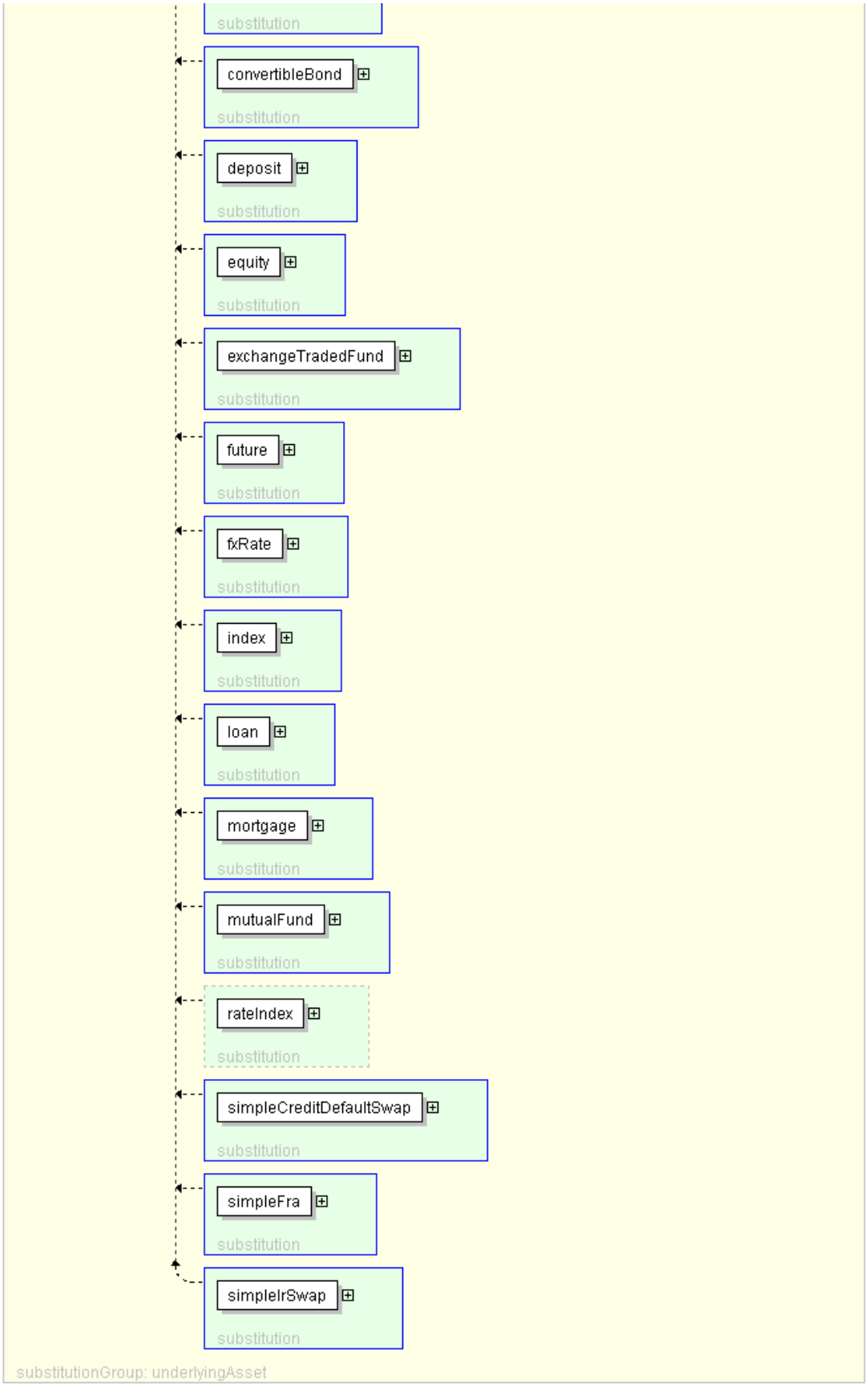
XML Instance Representation

```
<rateIndex
 id=" xsd:ID [0..1]">
```

```
    <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

    <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating a
    contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
    2002 Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail. In
    case of inconsistency between the terms of the simple product and those of the detailed definition,
    the values in the simple product override those in the detailed definition.'

    <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
    <term> Period </term> [1]
    'Specifies the term of the simple swap, e.g. 5Y.'

    <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the index pays, e.g. 6M.'

    <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the index.'

</rateIndex>
```

Diagram





Schema Component Representation

```
<xsd:element name="rateIndex" type=" RateIndex " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

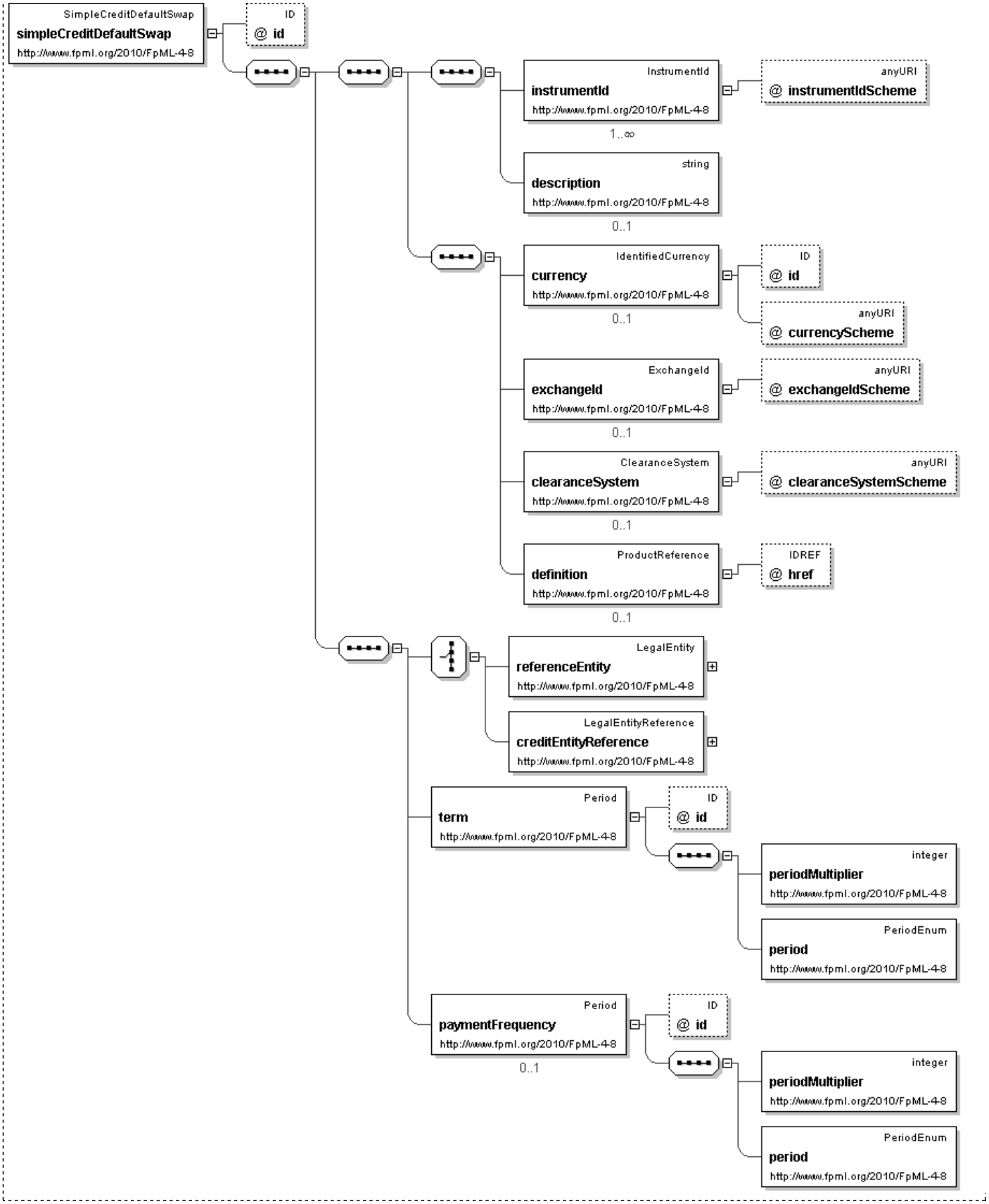
Element: simpleCreditDefaultSwap

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	simpleCreditDefaultSwap
Type	SimpleCreditDefaultSwap
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is a credit default swap.

Logical Diagram

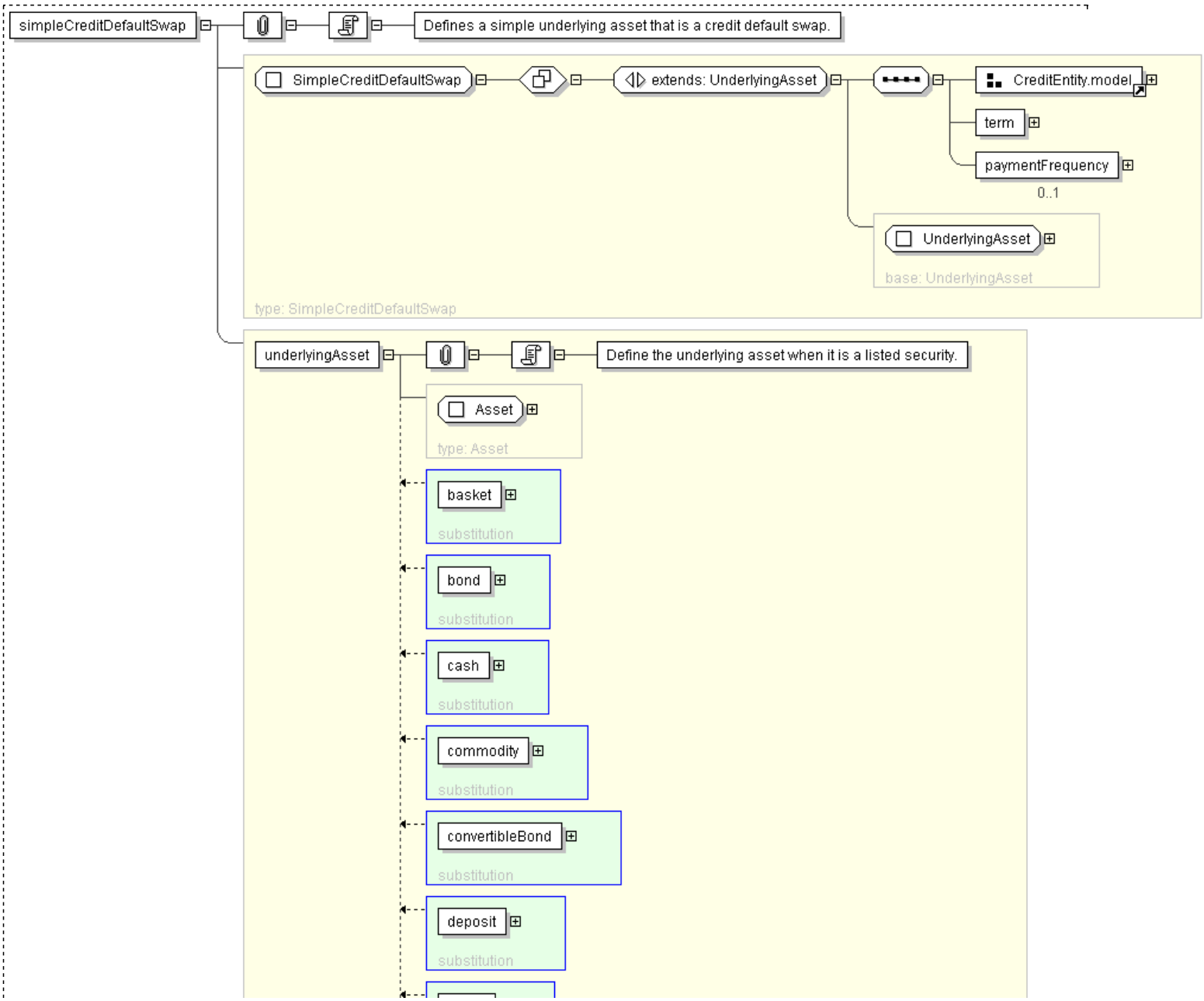


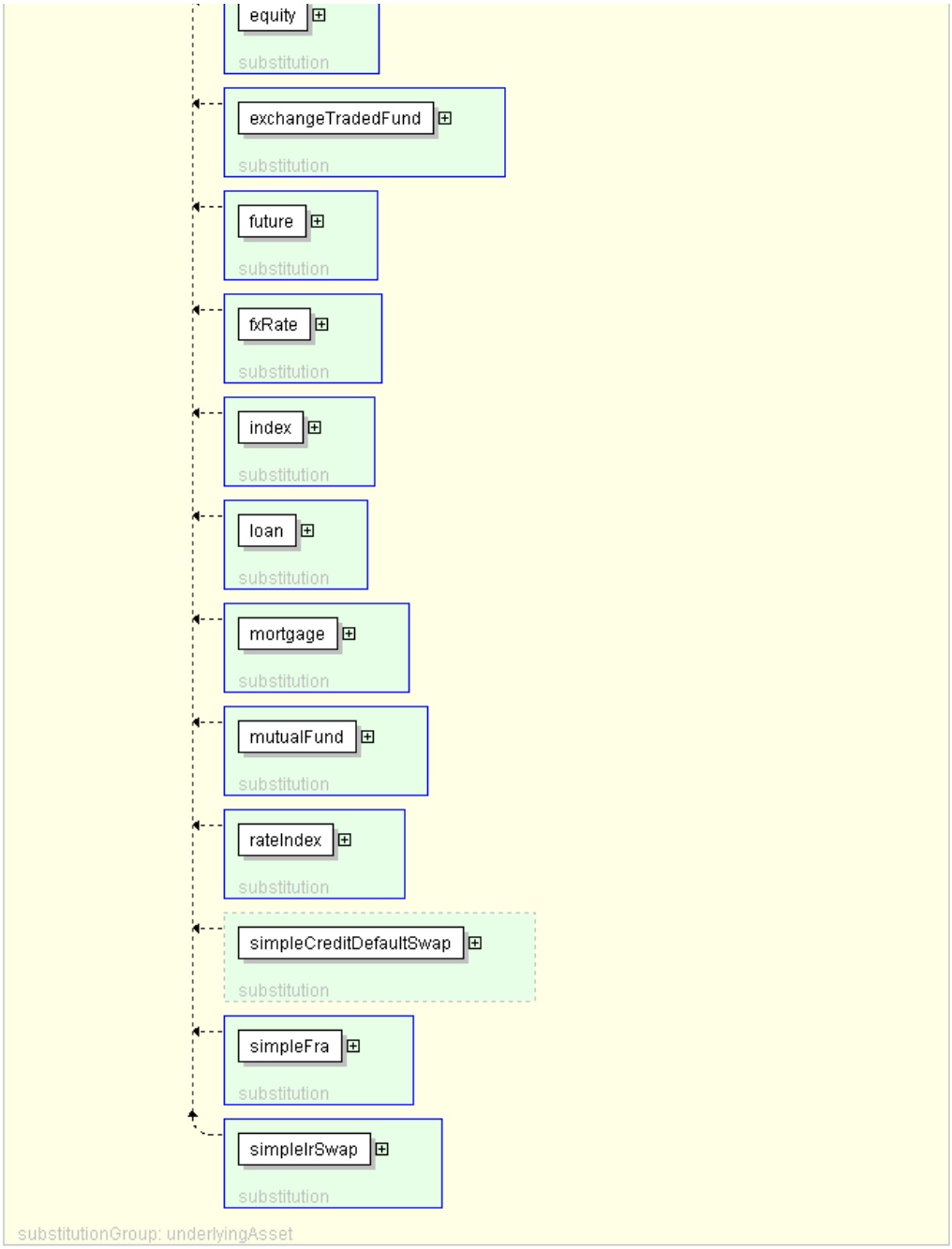
XML Instance Representation

```
<simpleCreditDefaultSwap
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
```

```
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating a
    contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002
    Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
    inconsistency between the terms of the simple product and those of the detailed definition, the values in
    the simple product override those in the detailed definition.'

    Start Choice [1]
       <referenceEntity> LegalEntity </referenceEntity> [1]
       'The entity for which this is defined.'

       <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
       'An XML reference a credit entity defined elsewhere in the document.'

    End Choice
    <term> Period </term> [1]
    'Specifies the term of the simple CD swap, e.g. 5Y.'

    <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the swap pays, e.g. 6M.'

</simpleCreditDefaultSwap>
```

Diagram





Schema Component Representation

```
<xsd:element name="simpleCreditDefaultSwap" type=" SimpleCreditDefaultSwap " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

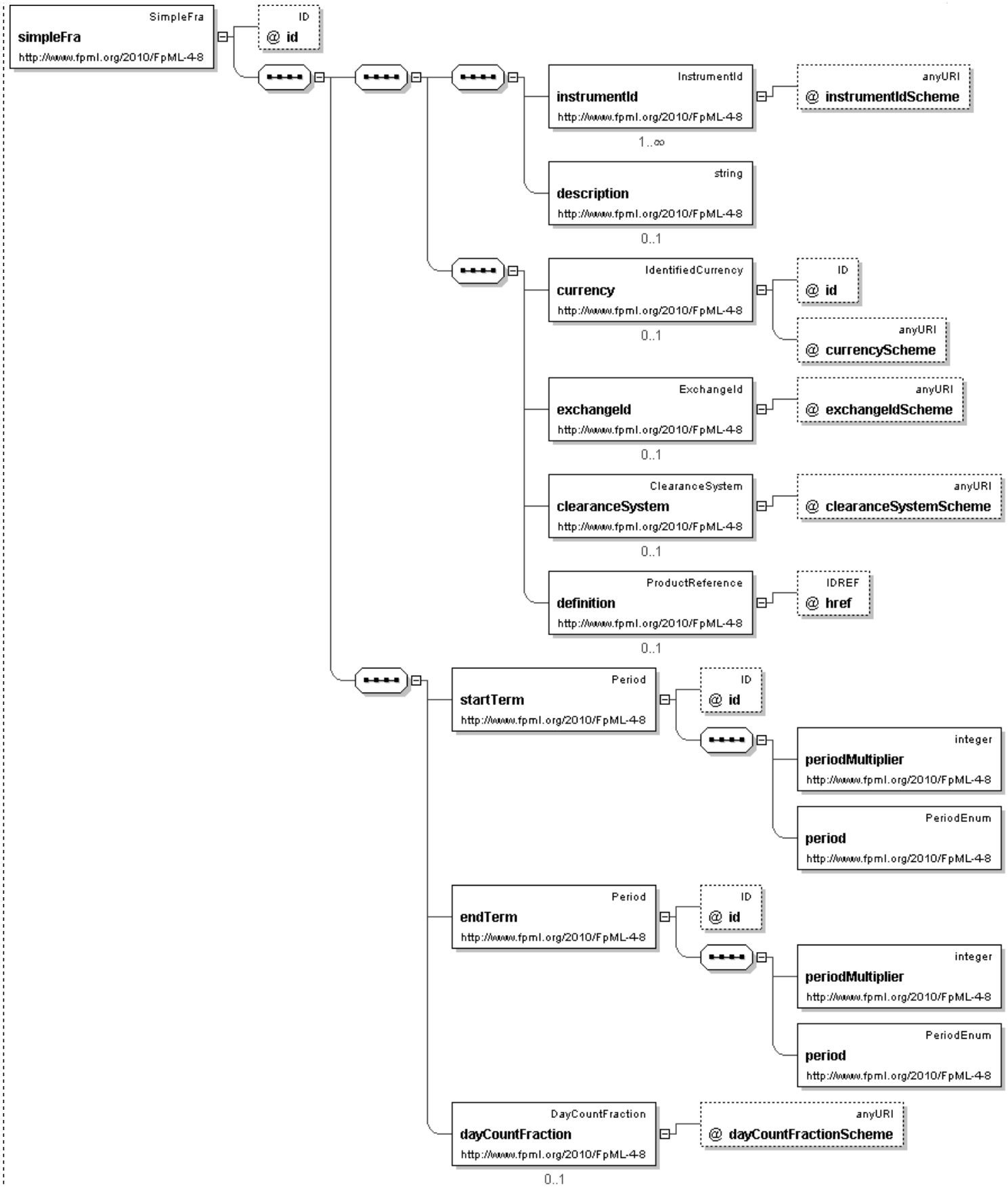
Element: simpleFra

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	simpleFra
Type	SimpleFra
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is a forward rate agreement.

Logical Diagram

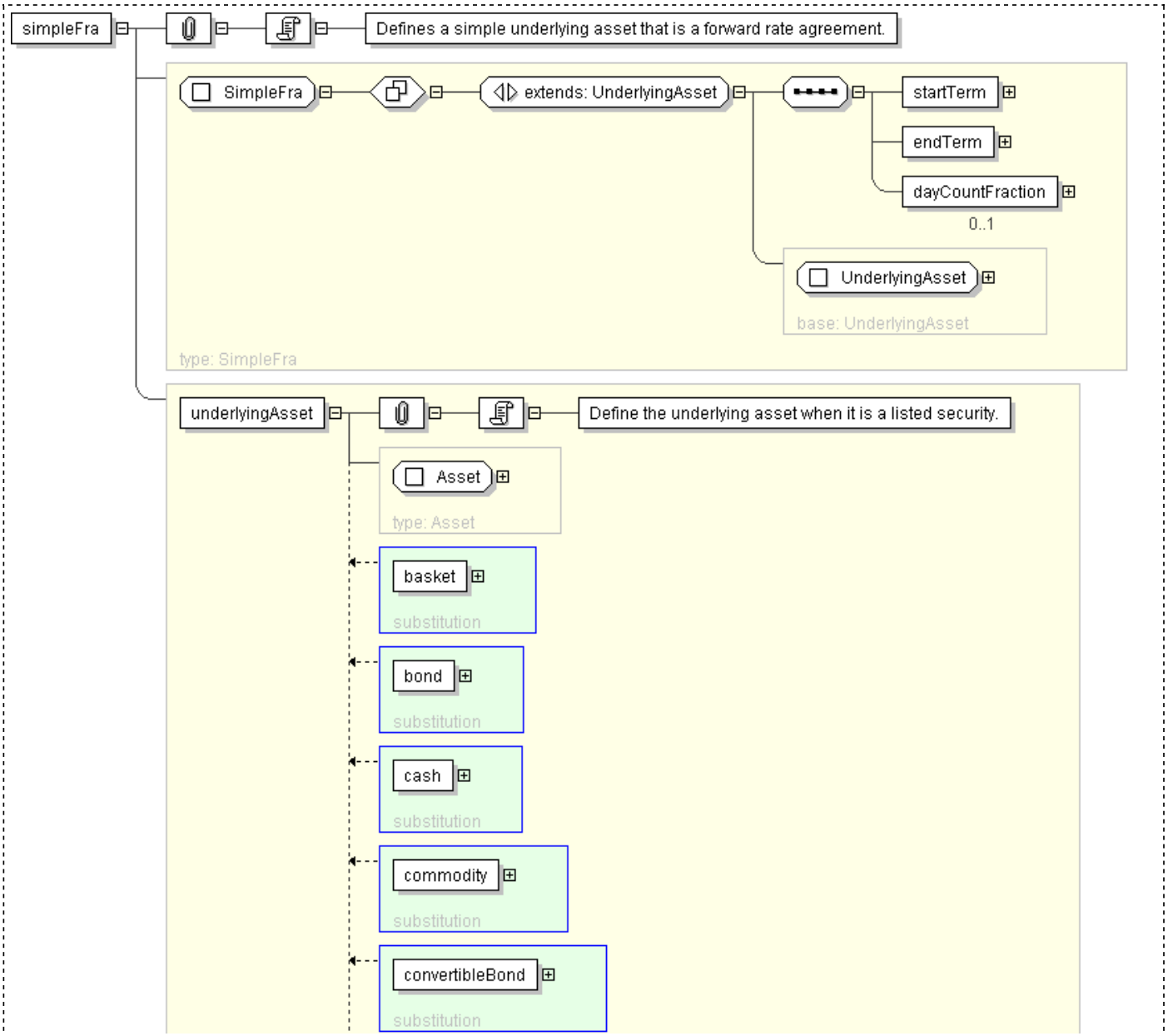


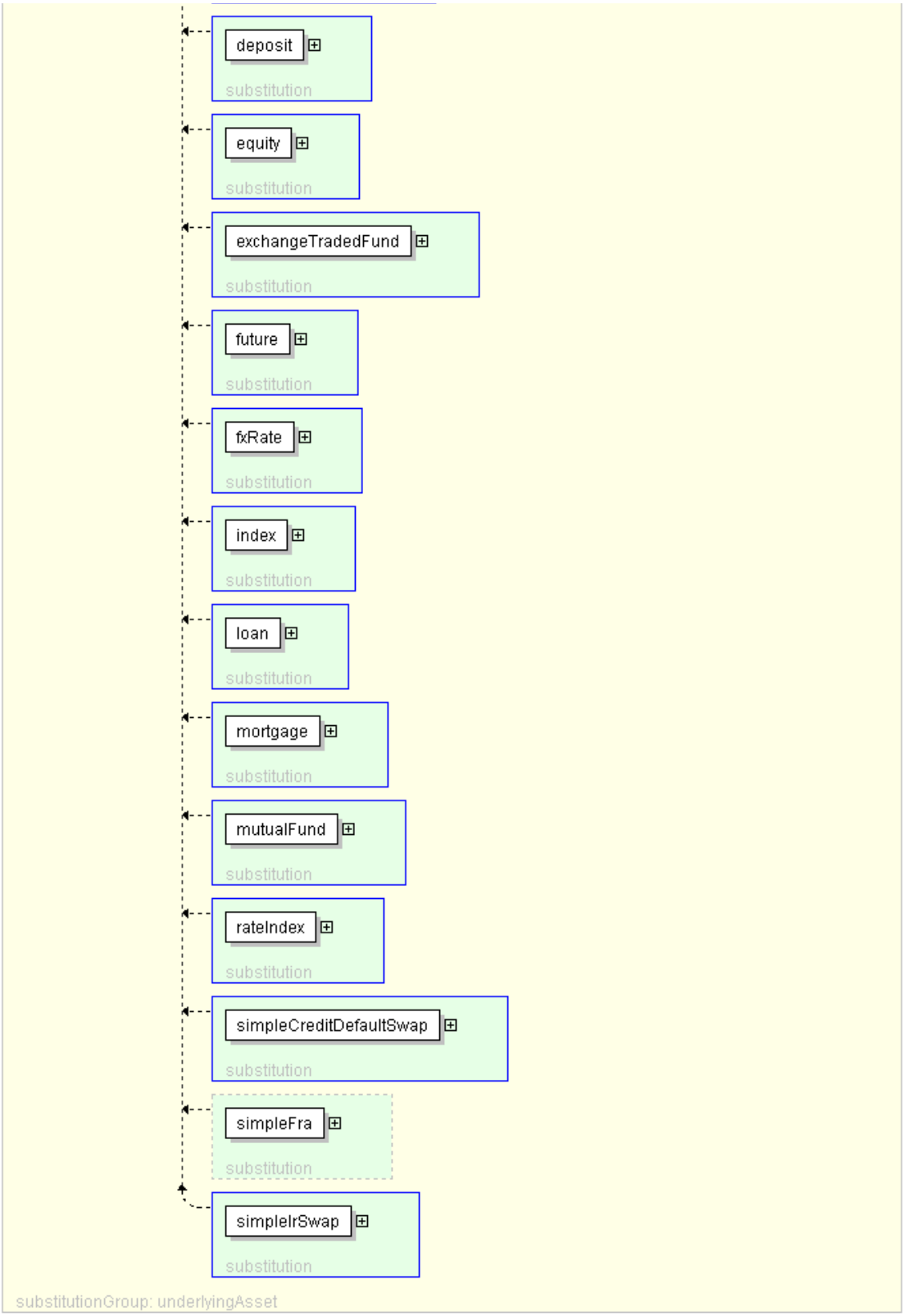
XML Instance Representation

```
<simpleFra
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'
```

```
    <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating
    a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
    2002 Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail.
    In case of inconsistency between the terms of the simple product and those of the detailed
    definition, the values in the simple product override those in the detailed definition.'

    <startTerm> Period </startTerm> [1]
    'Specifies the start term of the simple fra, e.g. 3M.'

    <endTerm> Period </endTerm> [1]
    'Specifies the end term of the simple fra, e.g. 9M.'

    <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the FRA.'

</simpleFra>
```

Diagram





Schema Component Representation

```
<xsd:element name="simpleFra" type=" SimpleFra " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

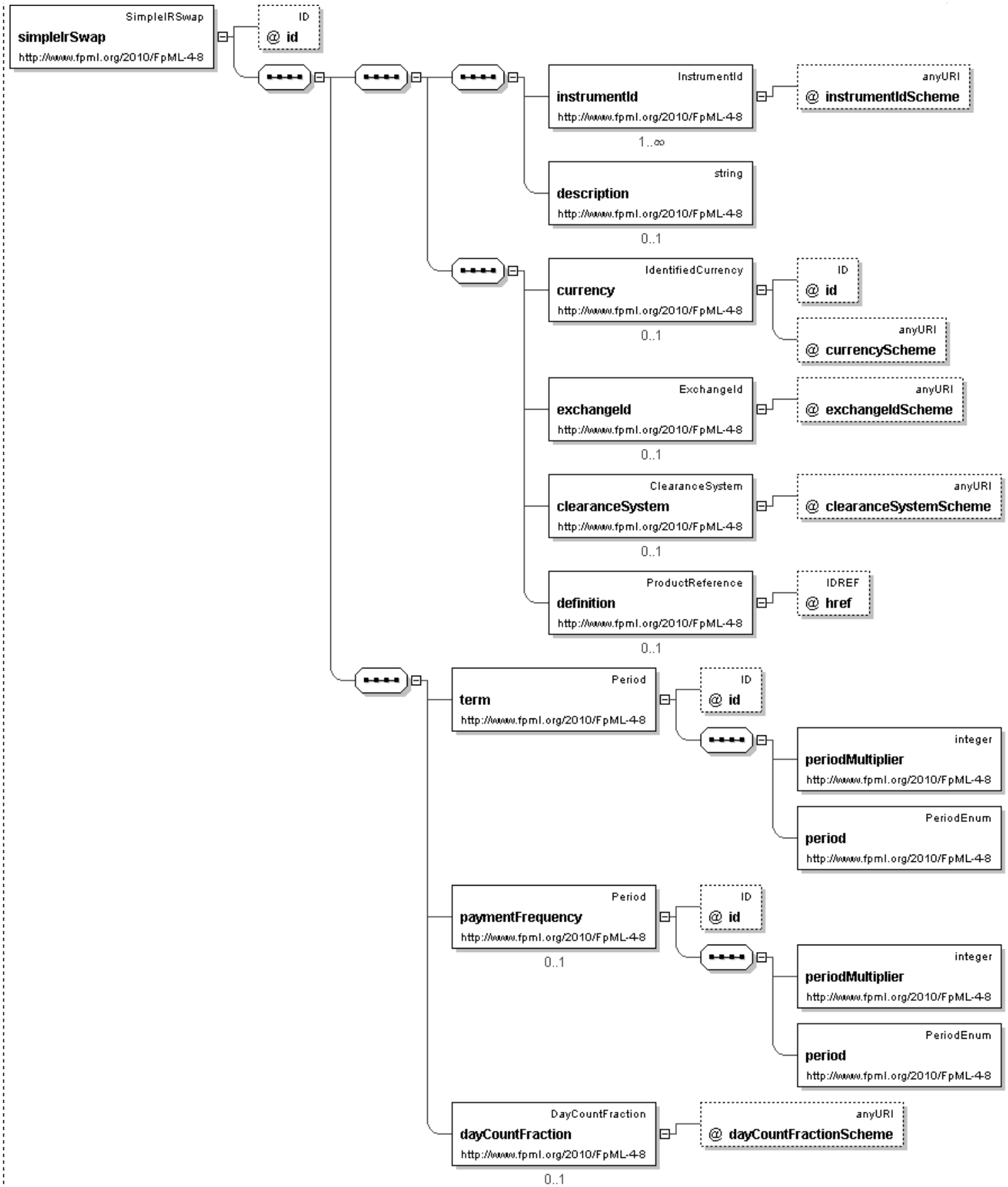
Element: simpleIrSwap

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - underlyingAsset

Name	simpleIrSwap
Type	SimpleIRSwap
Nillable	no
Abstract	no
Documentation	Defines a simple underlying asset that is a swap.

Logical Diagram

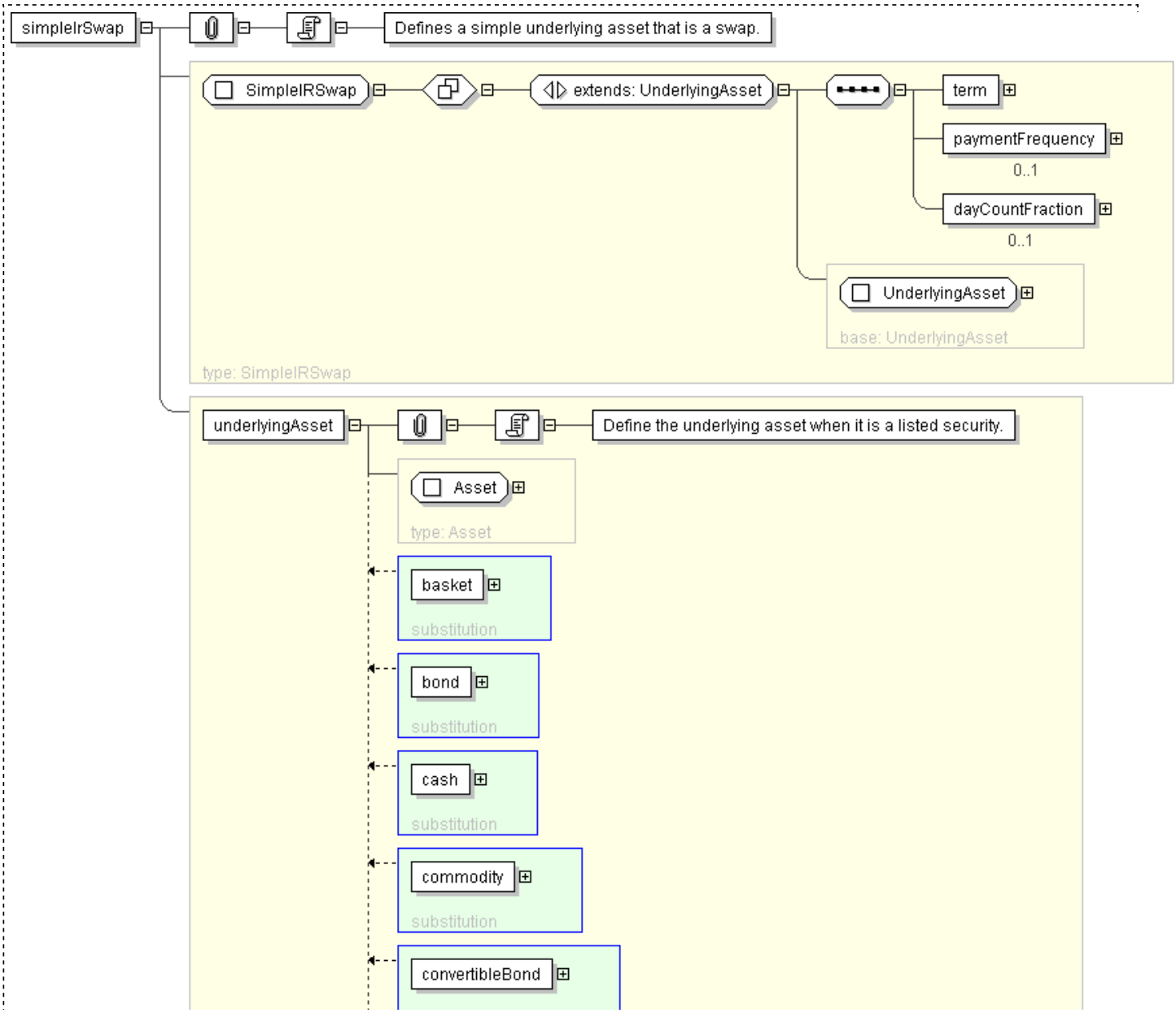


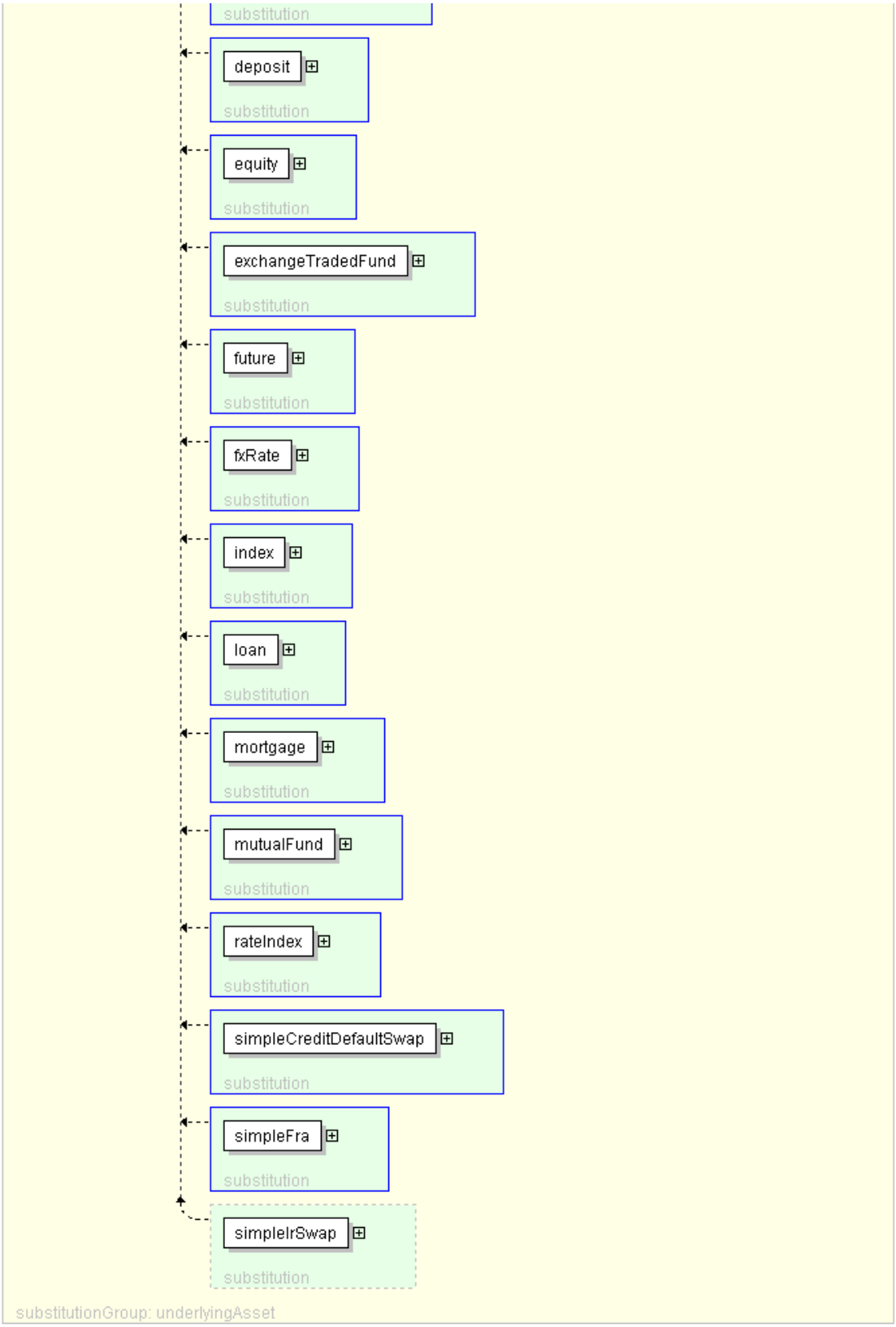
XML Instance Representation

```
<simpleIrSwap
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'
```

```
    <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

    <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

    <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating
    a contractural payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA
    2002 Equity Derivatives Definitions.'

    <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

    <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail.
    In case of inconsistency between the terms of the simple product and those of the detailed
    definition, the values in the simple product override those in the detailed definition.'

    <term> Period </term> [1]
    'Specifies the term of the simple swap, e.g. 5Y.'

    <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the swap pays, e.g. 6M.'

    <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the swap.'

</simpleIrSwap>
```

Diagram





Schema Component Representation

```
<xsd:element name="simpleIrSwap" type=" SimpleIRSwap " substitutionGroup="underlyingAsset"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: underlyingAsset

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - basket
  - bond
  - cash
  - commodity
  - convertibleBond
  - deposit
  - equity
  - exchangeTradedFund
  - future
  - fxRate
  - index
  - loan
  - mortgage
  - mutualFund
  - rateIndex
  - simpleCreditDefaultSwap
  - simpleFra
  - simpleIrSwap

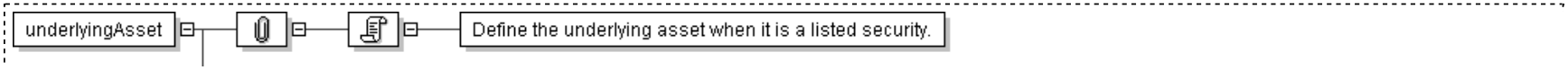
Name	underlyingAsset
Used by (from the same schema document)	Complex Type BasketConstituent , Complex Type SingleUnderlyer
Type	Asset
Nillable	no
Abstract	yes
Documentation	Define the underlying asset when it is a listed security.

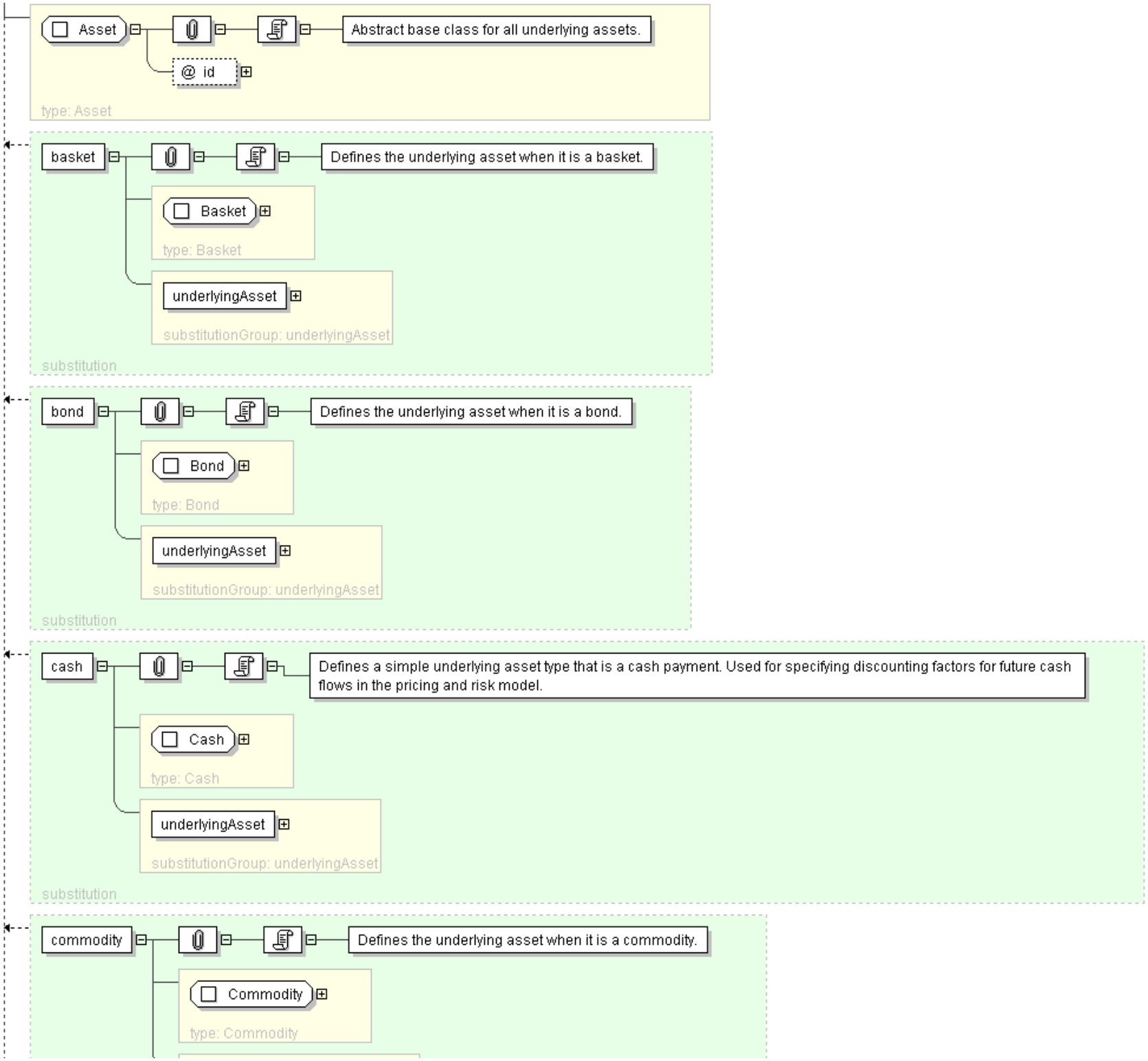
Logical Diagram

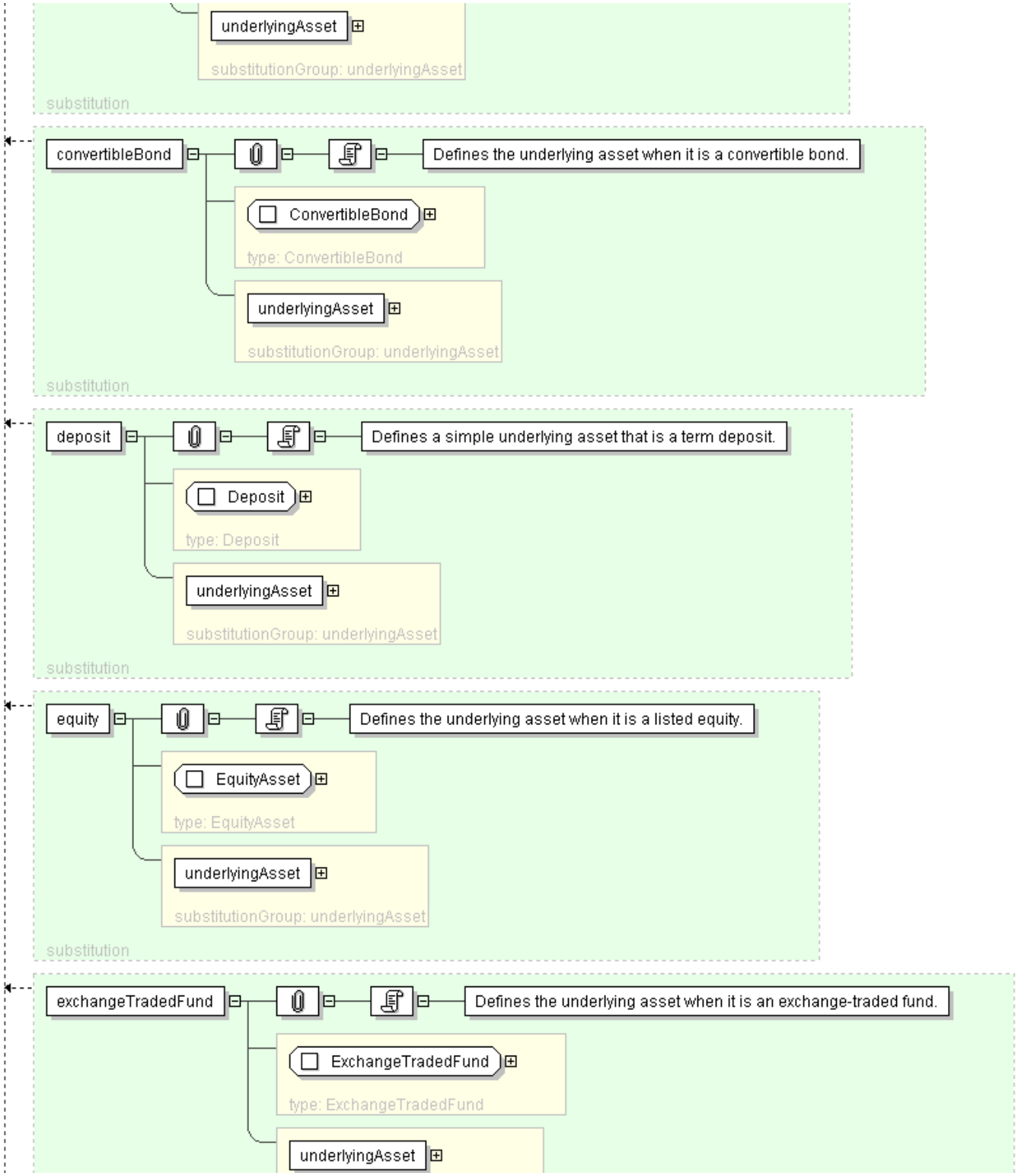
XML Instance Representation

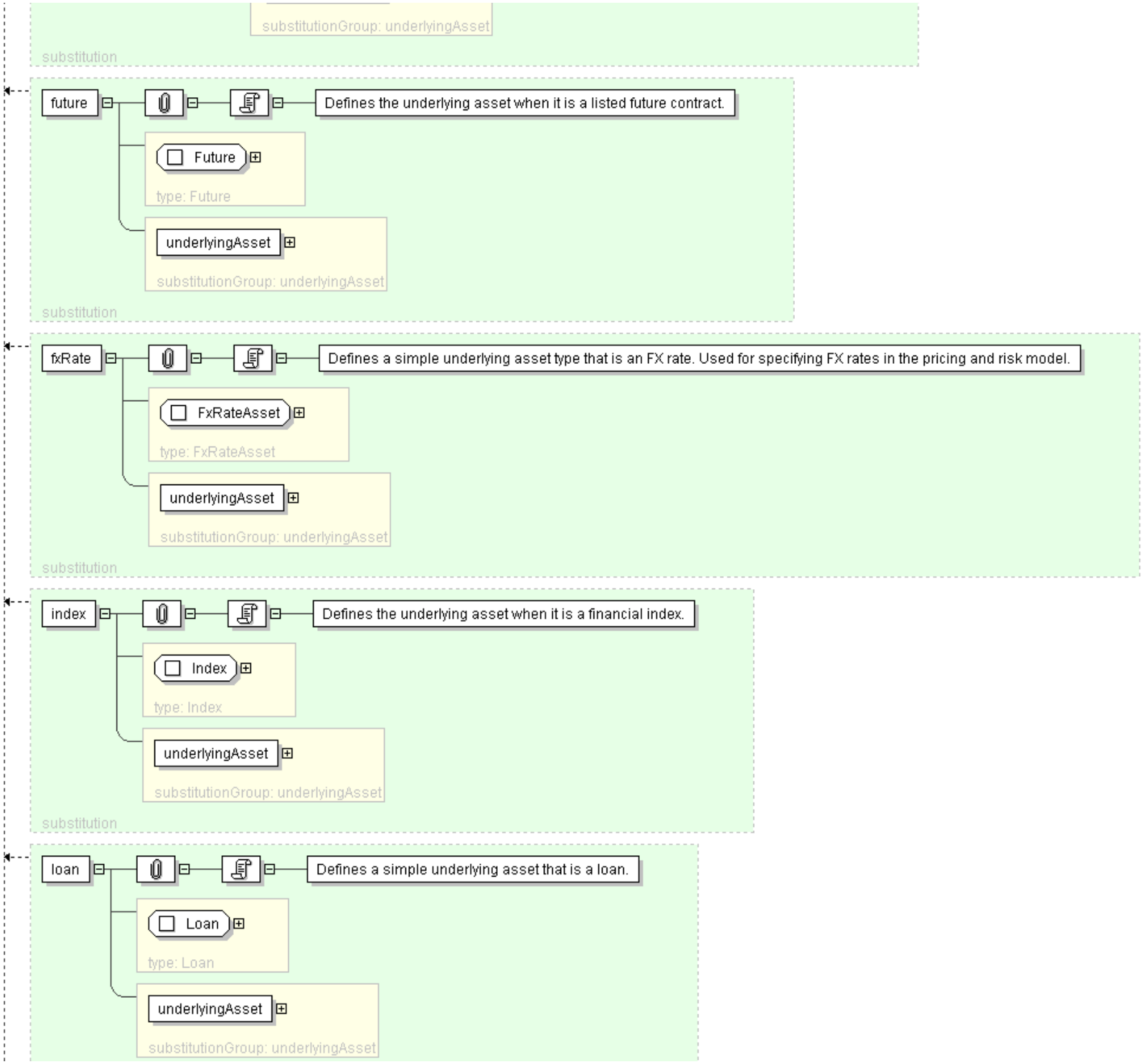
```
<underlyingAsset
 id=" xsd:ID [0..1]"/>
```

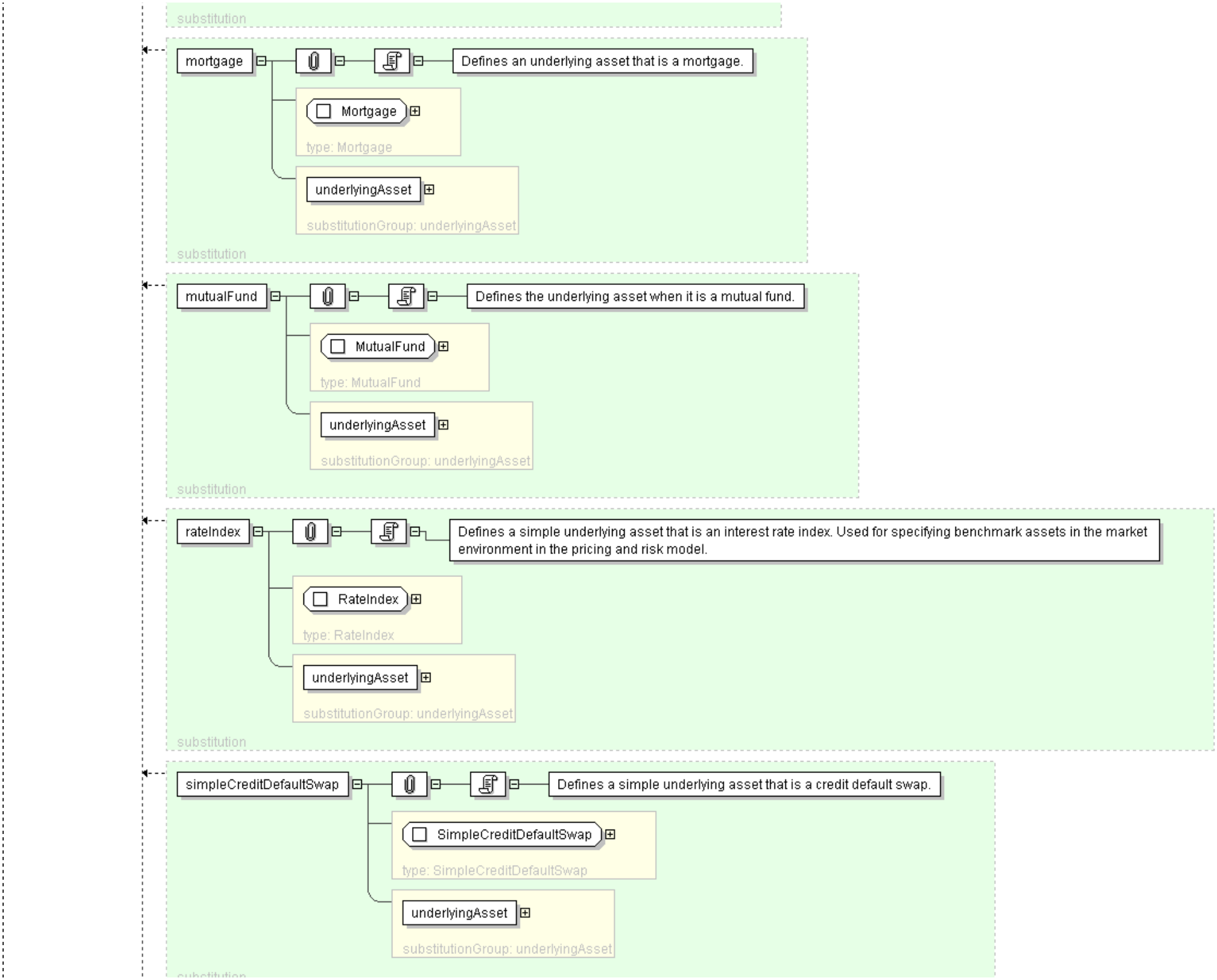
Diagram

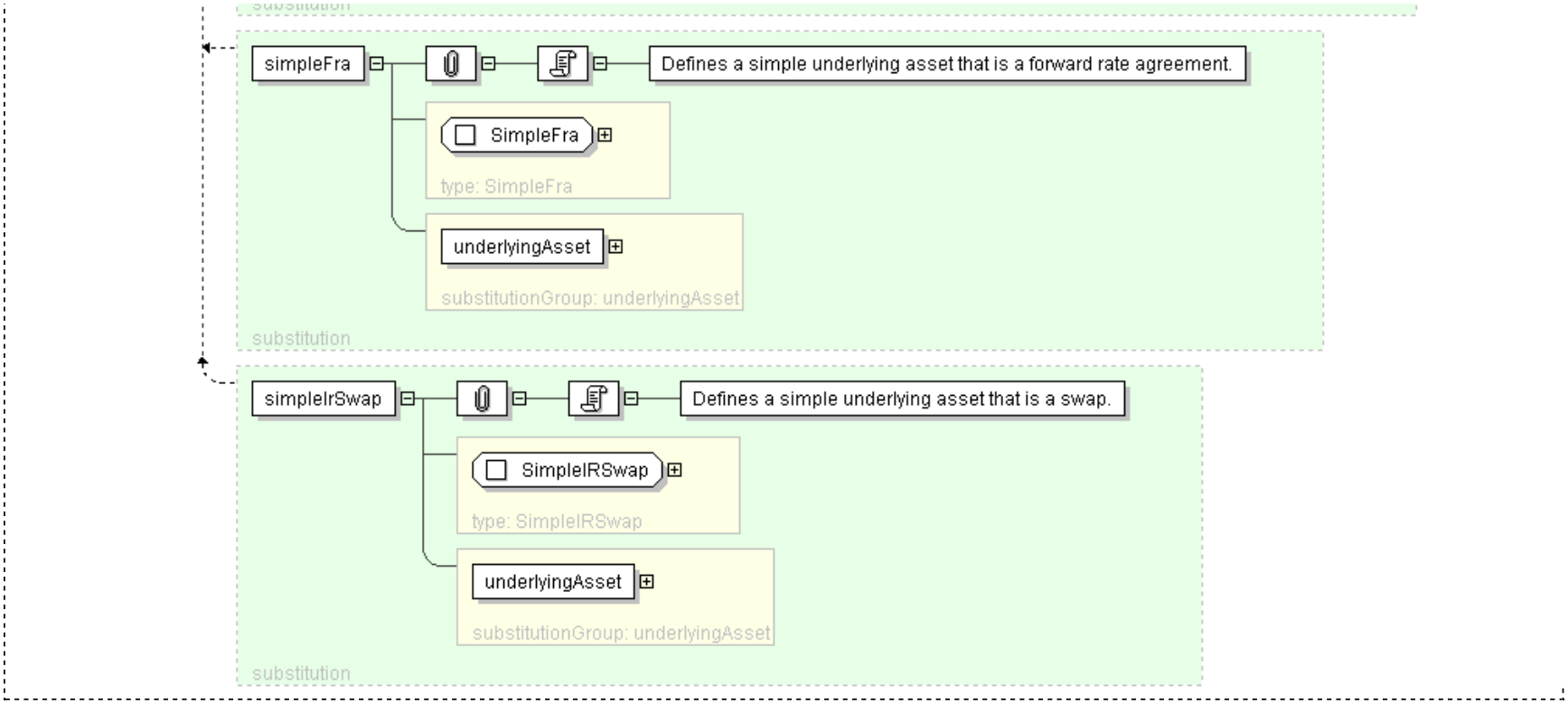



Asset
Abstract base class for all underlying assets.
@ id
type: Asset
basket
Defines the underlying asset when it is a basket.
Basket
type: Basket
underlyingAsset
substitutionGroup: underlyingAsset
substitution
bond
Defines the underlying asset when it is a bond.
Bond
type: Bond
underlyingAsset
substitutionGroup: underlyingAsset
substitution
cash
Defines a simple underlying asset type that is a cash payment. Used for specifying discounting factors for future cash flows in the pricing and risk model.
Cash
type: Cash
underlyingAsset
substitutionGroup: underlyingAsset
substitution
commodity
Defines the underlying asset when it is a commodity.
Commodity
type: Commodity


underlyingAsset
substitutionGroup: underlyingAsset
substitution
convertibleBond
Defines the underlying asset when it is a convertible bond.
ConvertibleBond
type: ConvertibleBond
underlyingAsset
substitutionGroup: underlyingAsset
substitution
deposit
Defines a simple underlying asset that is a term deposit.
Deposit
type: Deposit
underlyingAsset
substitutionGroup: underlyingAsset
substitution
equity
Defines the underlying asset when it is a listed equity.
EquityAsset
type: EquityAsset
underlyingAsset
substitutionGroup: underlyingAsset
substitution
exchangeTradedFund
Defines the underlying asset when it is an exchange-traded fund.
ExchangeTradedFund
type: ExchangeTradedFund
underlyingAsset


substitutionGroup: underlyingAsset
substitution
future
Defines the underlying asset when it is a listed future contract.
Future
type: Future
underlyingAsset
substitutionGroup: underlyingAsset
substitution
fxRate
Defines a simple underlying asset type that is an FX rate. Used for specifying FX rates in the pricing and risk model.
FxRateAsset
type: FxRateAsset
underlyingAsset
substitutionGroup: underlyingAsset
substitution
index
Defines the underlying asset when it is a financial index.
Index
type: Index
underlyingAsset
substitutionGroup: underlyingAsset
substitution
loan
Defines a simple underlying asset that is a loan.
Loan
type: Loan
underlyingAsset
substitutionGroup: underlyingAsset


substitution
mortgage
Defines an underlying asset that is a mortgage.
Mortgage
type: Mortgage
underlyingAsset
substitutionGroup: underlyingAsset
substitution
mutualFund
Defines the underlying asset when it is a mutual fund.
MutualFund
type: MutualFund
underlyingAsset
substitutionGroup: underlyingAsset
substitution
rateIndex
Defines a simple underlying asset that is an interest rate index. Used for specifying benchmark assets in the market environment in the pricing and risk model.
RateIndex
type: RateIndex
underlyingAsset
substitutionGroup: underlyingAsset
substitution
simpleCreditDefaultSwap
Defines a simple underlying asset that is a credit default swap.
SimpleCreditDefaultSwap
type: SimpleCreditDefaultSwap
underlyingAsset
substitutionGroup: underlyingAsset



Schema Component Representation

```
<xsd:element name="underlyingAsset" type=" Asset " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BasketIdentifier.model

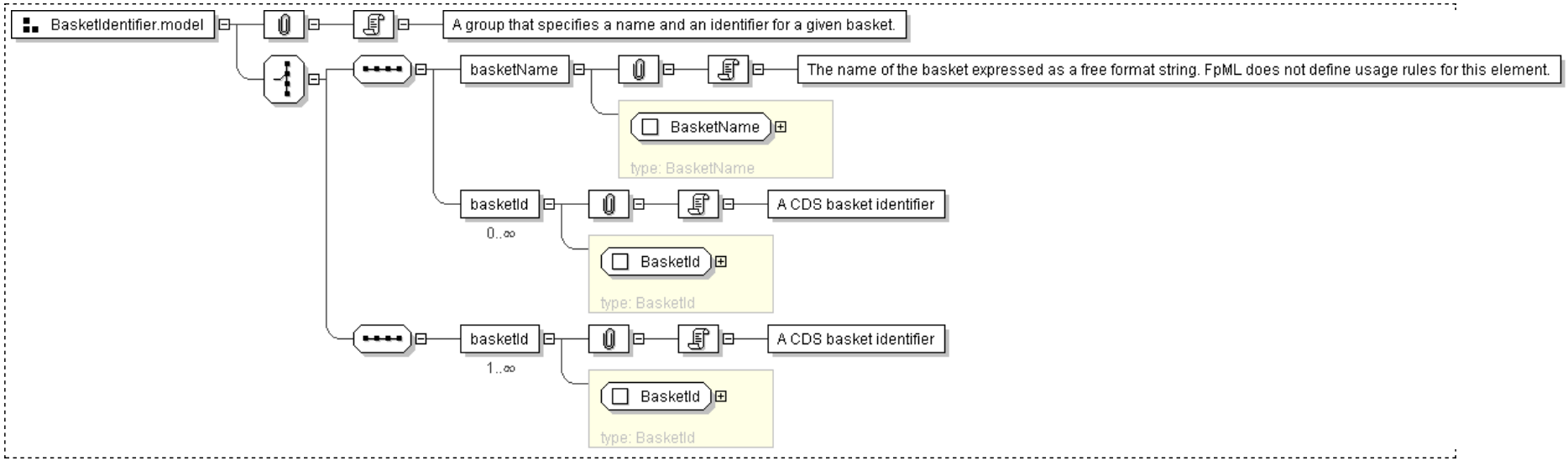
[Table of contents]

Name	BasketIdentifier.model
Used by (from the same schema document)	Complex Type Basket
Documentation	A group that specifies a name and an identifier for a given basket.

XML Instance Representation

```
Start Choice [1]
   <basketName> BasketName </basketName> [1]
   'The name of the basket expressed as a free format string. FpML does not define usage rules for this element.'

   <basketId> BasketId </basketId> [0..*]
   'A CDS basket identifier'

   <basketId> BasketId </basketId> [1..*]
   'A CDS basket identifier'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="BasketIdentifier.model">
   <xsd:choice>
      <xsd:sequence>
         <xsd:element name="basketName" type=" BasketName "/>
         <xsd:element name="basketId" type=" BasketId " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:sequence>
         <xsd:element name="basketId" type=" BasketId " maxOccurs="unbounded"/>
      </xsd:sequence>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BondCalculation.model

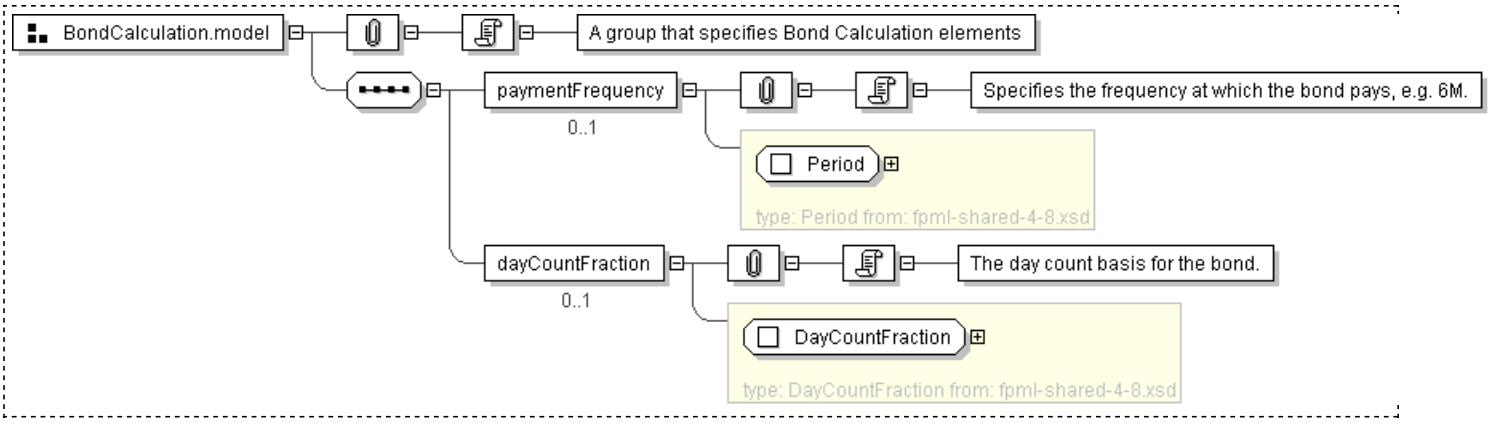
[Table of contents]

Name	BondCalculation.model
Used by (from the same schema document)	Complex Type Bond , Complex Type Mortgage
Documentation	A group that specifies Bond Calculation elements

XML Instance Representation

```
<paymentFrequency> Period </paymentFrequency> [0..1]
'Specifies the frequency at which the bond pays, e.g. 6M.'

<dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
'The day count basis for the bond.'
```

Diagram



Schema Component Representation

```
<xsd:group name="BondCalculation.model">
  <xsd:sequence>
    <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
    <xsd:element name="dayCountFraction" type=" DayCountFraction " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BondChoice.model

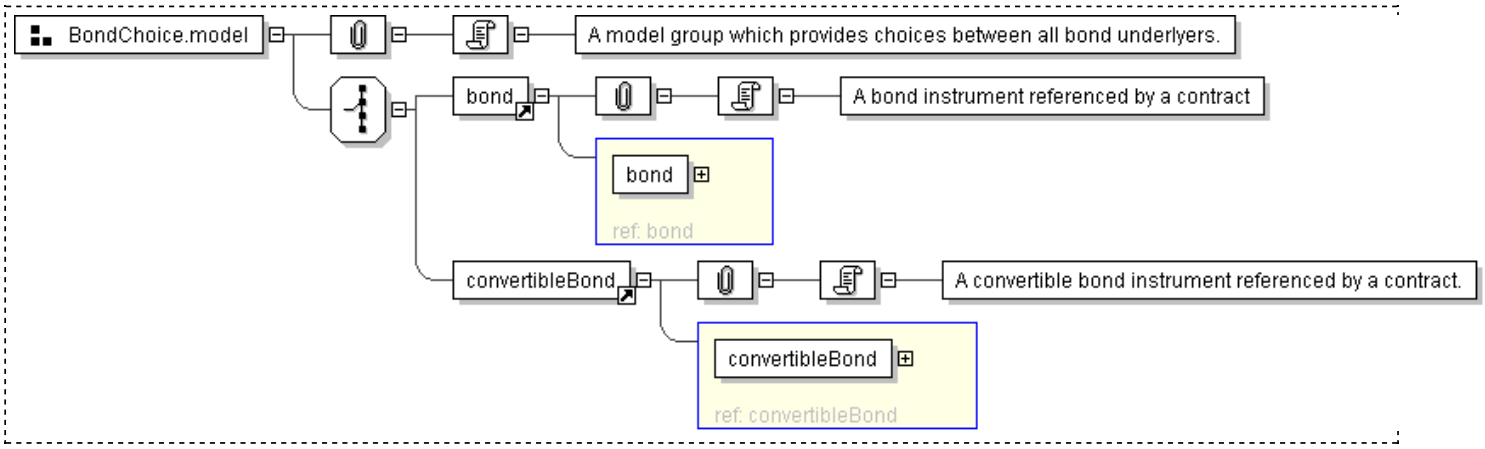
[Table of contents]

Name	BondChoice.model
Documentation	A model group which provides choices between all bond underlyers.

XML Instance Representation

```
Start Choice [1]
   <bond> ... </bond> [1]
   'A bond instrument referenced by a contract'

   <convertibleBond> ... </convertibleBond> [1]
   'A convertible bond instrument referenced by a contract.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="BondChoice.model">
   <xsd:choice>
      <xsd:element ref=" bond "/>
      <xsd:element ref=" convertibleBond "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BondContent.model

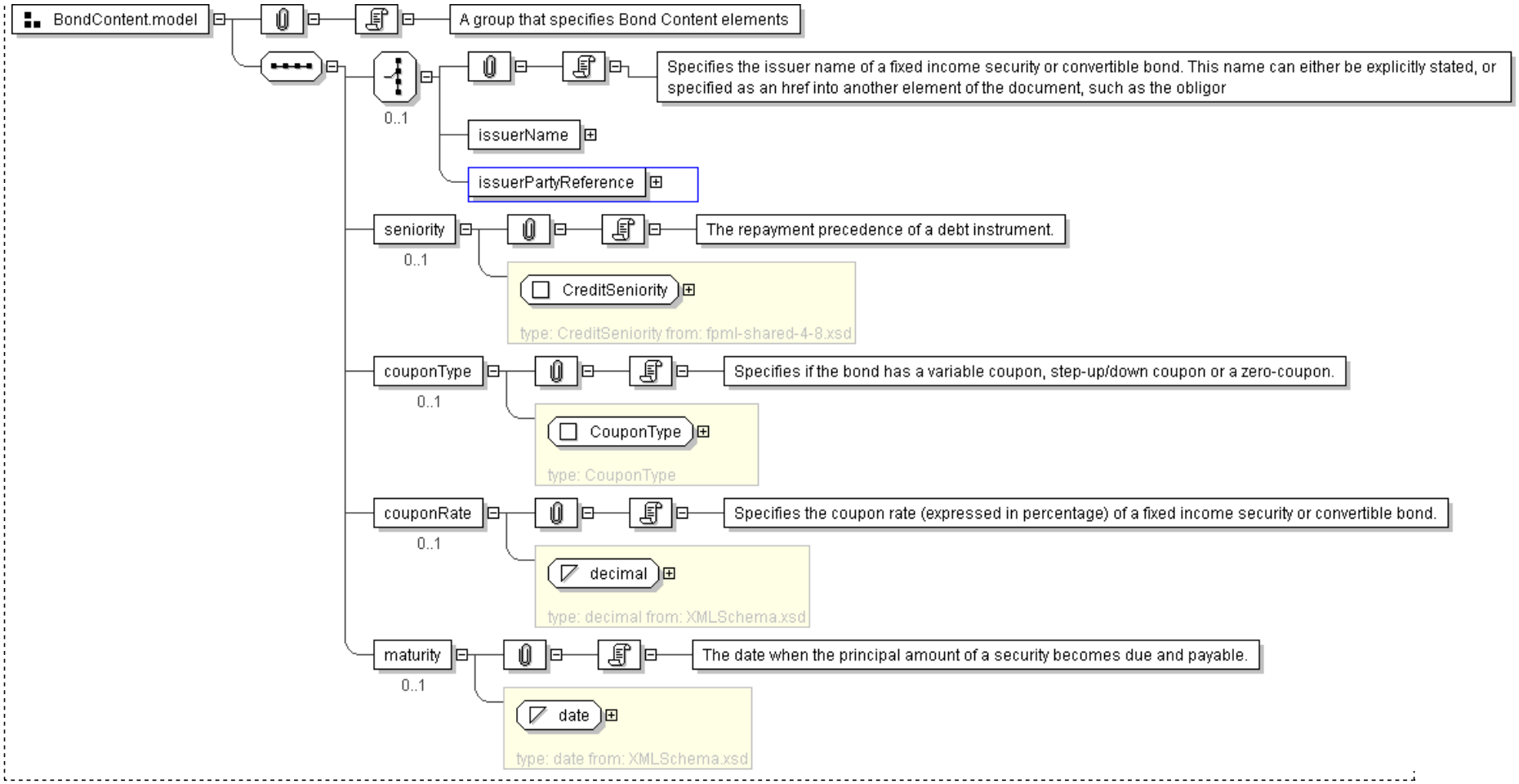
[Table of contents]

Name	BondContent.model
Used by (from the same schema document)	Complex Type Bond , Complex Type Mortgage
Documentation	A group that specifies Bond Content elements

XML Instance Representation

```
Start Choice [0..1]
'Specifies the issuer name of a fixed income security or convertible bond. This name can either be explicitly stated, or specified
as an href into another element of the document, such as the obligor'


   <issuerName> xsd:string </issuerName> [1]
   <issuerPartyReference> PartyReference </issuerPartyReference> [1]
End Choice
<seniority> CreditSeniority </seniority> [0..1]
'The repayment precedence of a debt instrument.'

<couponType> CouponType </couponType> [0..1]
'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-coupon.'

<couponRate> xsd:decimal </couponRate> [0..1]
'Specifies the coupon rate (expressed in percentage) of a fixed income security or convertible bond.'

<maturity> xsd:date </maturity> [0..1]
'The date when the principal amount of a security becomes due and payable.'
```

Diagram



Schema Component Representation

```
<xsd:group name="BondContent.model">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="issuerName" type=" xsd:string "/>
      <xsd:element name="issuerPartyReference" type=" PartyReference "/>
    </xsd:choice>
    <xsd:element name="seniority" type=" CreditSeniority " minOccurs="0"/>
    <xsd:element name="couponType" type=" CouponType " minOccurs="0"/>
    <xsd:element name="couponRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="maturity" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityProduct.model

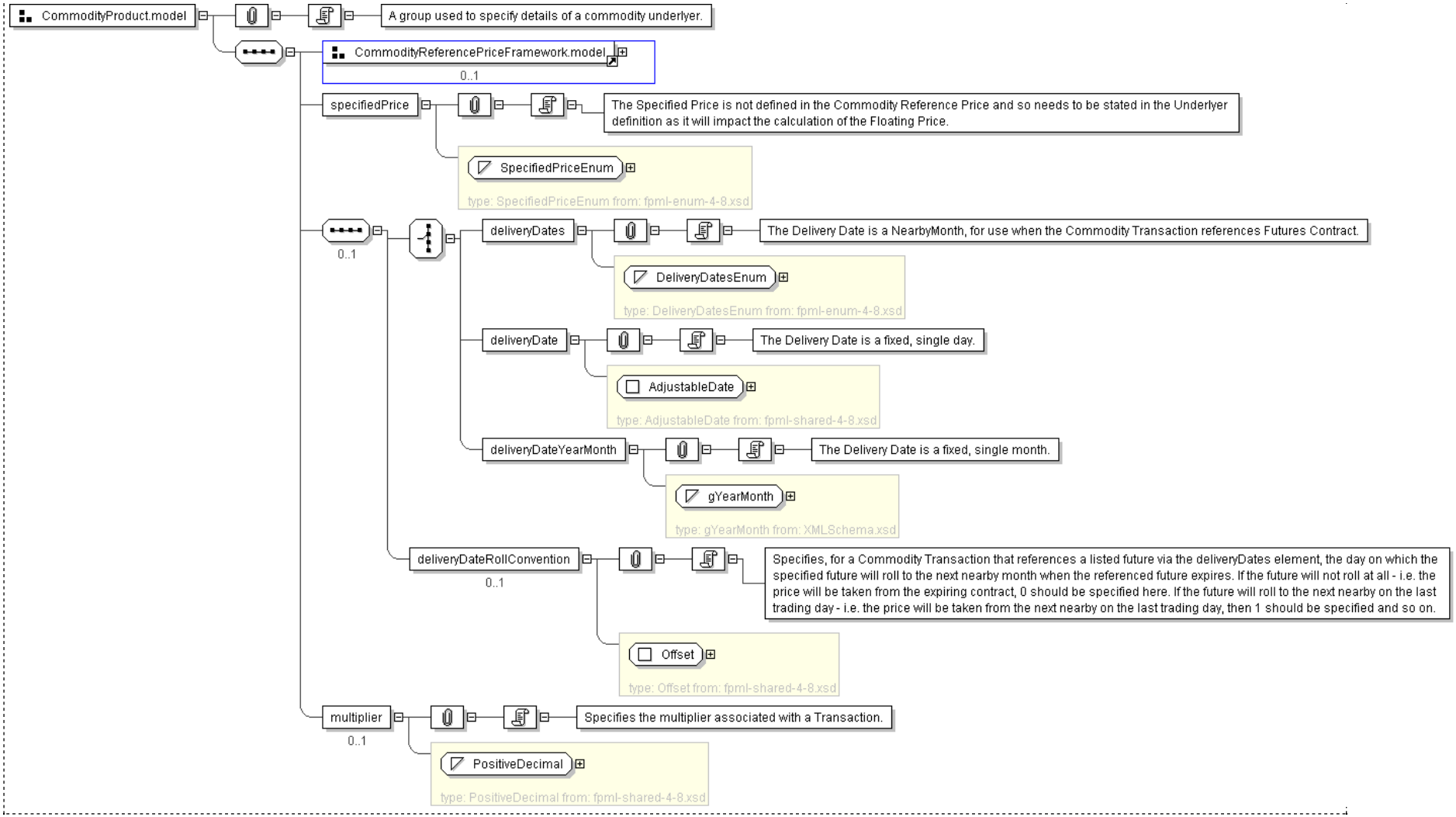
[Table of contents]

Name	CommodityProduct.model
Used by (from the same schema document)	Complex Type Commodity
Documentation	A group used to specify details of a commodity underlyer.

XML Instance Representation

```
Start Group: CommodityReferencePriceFramework.model [0..1]
   <commodityBase> CommodityBase </commodityBase> [1]
   'A coding scheme value to identify the base type of the commodity being traded. Where possible, this should follow the naming convention used in the 2005 ISDA
   Commodity Definitions. For example, \'Oil\'.'

   <commodityDetails> CommodityDetails </commodityDetails> [1]
   'A coding scheme value to identify the commodity being traded more specifically. Where possible, this should follow the naming convention used in the 2005 ISDA
   Commodity Definitions. For example, \'Brent\'.'

   <unit> QuantityUnit </unit> [1]
   'A coding scheme value to identify the unit in which the undelryer is denominated. Where possible, this should follow the naming convention used in the 2005
   ISDA Commodity Definitions.'

   <currency> Currency </currency> [1]
   'The currency in which the Commodity Reference Price is published.'

   Start Choice [1]
      <exchangeId> ExchangeId </exchangeId> [1]
      'For those commodities being traded with reference to the price of a listed future, the exchange where that future is listed should be specified here.'

      <publication> InformationSource </publication> [1]
      'For those commodities being traded with reference to a price distributed by a publication, that publication should be specified here.'

   End Choice
End Group: CommodityReferencePriceFramework.model
<specifiedPrice> SpecifiedPriceEnum </specifiedPrice> [1]
'The Specified Price is not defined in the Commodity Reference Price and so needs to be stated in the Underlyer definition as it will impact the calculation of
the Floating Price.'

Start Sequence [0..1]
   Start Choice [1]
      <deliveryDates> DeliveryDatesEnum </deliveryDates> [1]
      'The Delivery Date is a NearbyMonth, for use when the Commodity Transaction references Futures Contract.'

      <deliveryDate> AdjustableDate </deliveryDate> [1]
      'The Delivery Date is a fixed, single day.'

      <deliveryDateYearMonth> xsd:gYearMonth </deliveryDateYearMonth> [1]
      'The Delivery Date is a fixed, single month.'

   End Choice
   <deliveryDateRollConvention> Offset </deliveryDateRollConvention> [0..1]
   'Specifies, for a Commodity Transaction that references a listed future via the deliveryDates element, the day on which the specified future will roll to the
   next nearby month when the referenced future expires. If the future will not roll at all - i.e. the price will be taken from the expiring contract, 0 should
   be specified here. If the future will roll to the next nearby on the last trading day - i.e. the price will be taken from the next nearby on the last trading
   day, then 1 should be specified and so on.'

End Sequence
<multiplier> PositiveDecimal </multiplier> [0..1]
'Specifies the multiplier associated with a Transaction.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityProduct.model">
  <xsd:sequence>
    <xsd:group ref=" CommodityReferencePriceFramework.model " minOccurs="0"/>
    <xsd:element name="specifiedPrice" type=" SpecifiedPriceEnum "/>
    <xsd:sequence minOccurs="0">
      <xsd:choice>
        <xsd:element name="deliveryDates" type=" DeliveryDatesEnum "/>
        <xsd:element name="deliveryDate" type=" AdjustableDate "/>
        <xsd:element name="deliveryDateYearMonth" type=" xsd:gYearMonth "/>
      </xsd:choice>
      <xsd:element name="deliveryDateRollConvention" type=" Offset " minOccurs="0"/>
    </xsd:sequence>
    <xsd:element name="multiplier" type=" PositiveDecimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityReferencePriceFramework.model

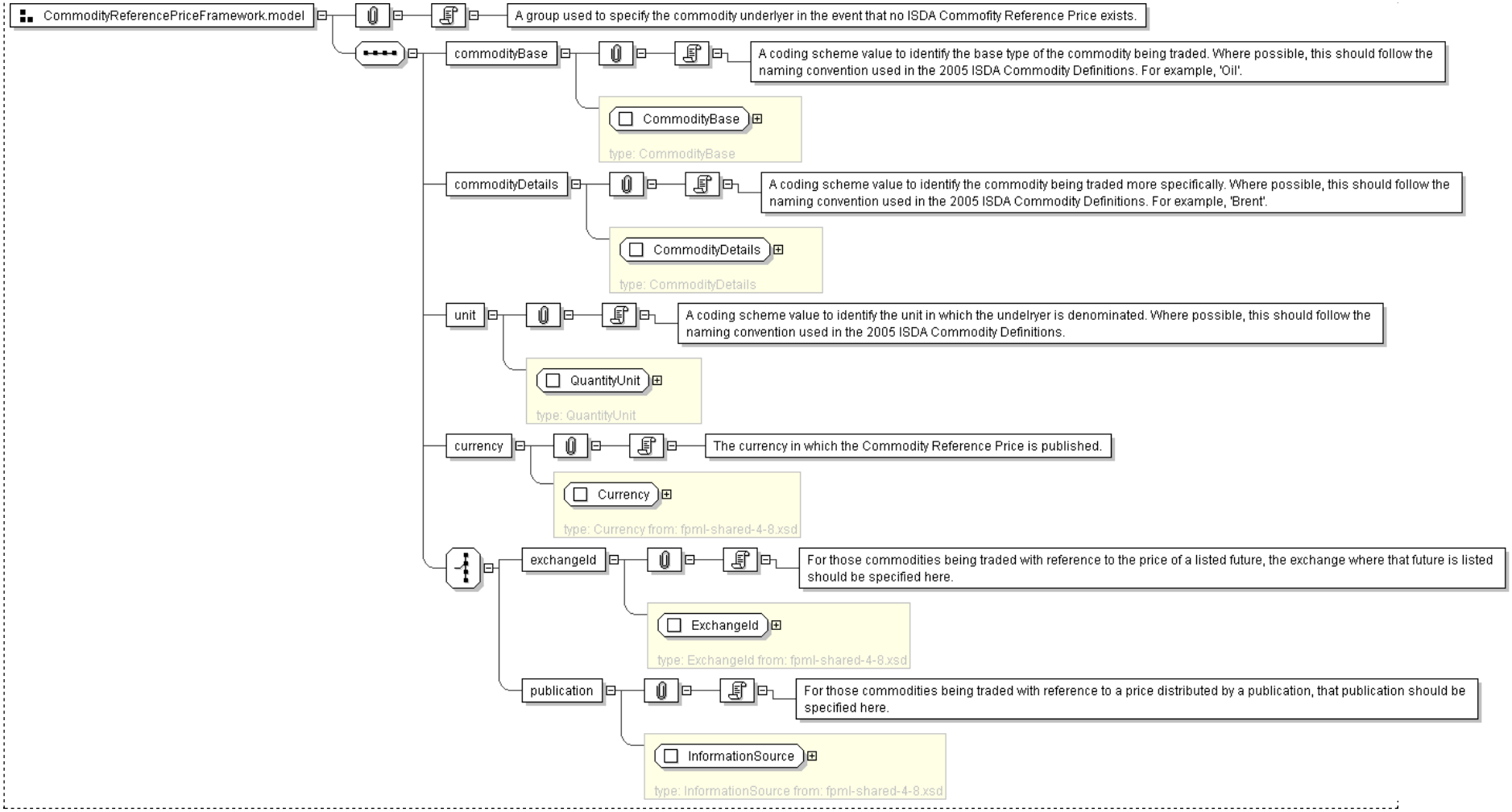
[Table of contents]

Name	CommodityReferencePriceFramework.model
Used by (from the same schema document)	Model Group CommodityProduct.model
Documentation	A group used to specify the commodity underlyer in the event that no ISDA Commofity Reference Price exists.

XML Instance Representation

```
<commodityBase> CommodityBase </commodityBase> [1]
'A coding scheme value to identify the base type of the commodity being traded. Where possible, this should follow the naming convention used in the 2005 ISDA
Commodity Definitions. For example, \'Oil\'.'

<commodityDetails> CommodityDetails </commodityDetails> [1]
'A coding scheme value to identify the commodity being traded more specifically. Where possible, this should follow the naming convention used in the 2005 ISDA
Commodity Definitions. For example, \'Brent\'.'

<unit> QuantityUnit </unit> [1]
'A coding scheme value to identify the unit in which the undelryer is denominated. Where possible, this should follow the naming convention used in the 2005 ISDA
Commodity Definitions.'

<currency> Currency </currency> [1]
'The currency in which the Commodity Reference Price is published.'

Start Choice [1]
   <exchangeId> ExchangeId </exchangeId> [1]
   'For those commodities being traded with reference to the price of a listed future, the exchange where that future is listed should be specified here.'

   <publication> InformationSource </publication> [1]
   'For those commodities being traded with reference to a price distributed by a publication, that publication should be specified here.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityReferencePriceFramework.model">
  <xsd:sequence>
    <xsd:element name="commodityBase" type=" CommodityBase "/>
    <xsd:element name="commodityDetails" type=" CommodityDetails "/>
    <xsd:element name="unit" type=" QuantityUnit "/>
    <xsd:element name="currency" type=" Currency "/>
    <xsd:choice>
      <xsd:element name="exchangeId" type=" ExchangeId "/>
      <xsd:element name="publication" type=" InformationSource "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CreditEntity.model

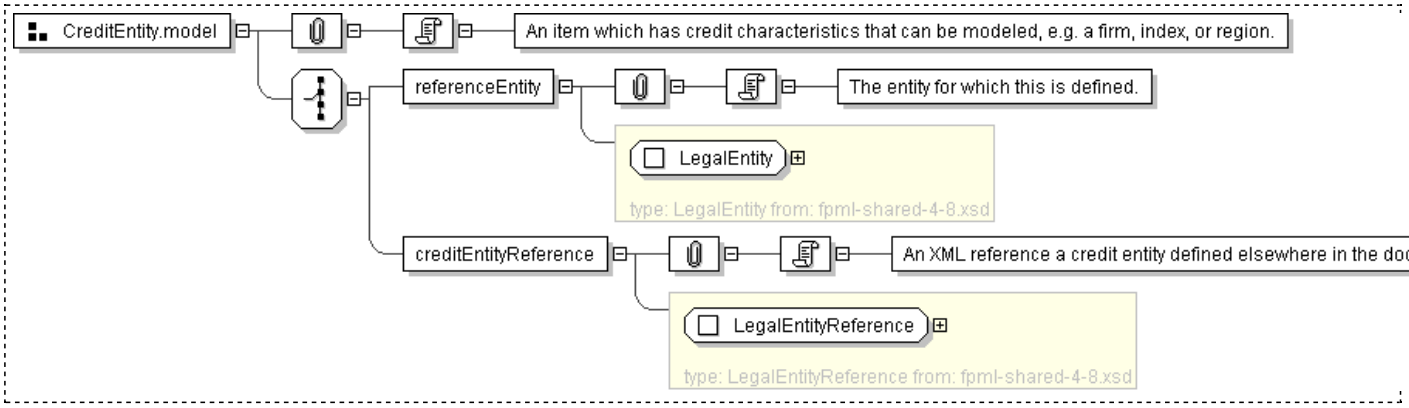
[Table of contents]

Name	CreditEntity.model
Used by (from the same schema document)	Complex Type SimpleCreditDefaultSwap
Documentation	An item which has credit characteristics that can be modeled, e.g. a firm, index, or region.

XML Instance Representation

```
Start Choice [1]
   <referenceEntity> LegalEntity </referenceEntity> [1]
   'The entity for which this is defined.'

   <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
   'An XML reference a credit entity defined elsewhere in the document.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CreditEntity.model">
   <xsd:choice>
      <xsd:element name="referenceEntity" type=" LegalEntity "/>
      <xsd:element name="creditEntityReference" type=" LegalEntityReference "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: EquityPrice.model

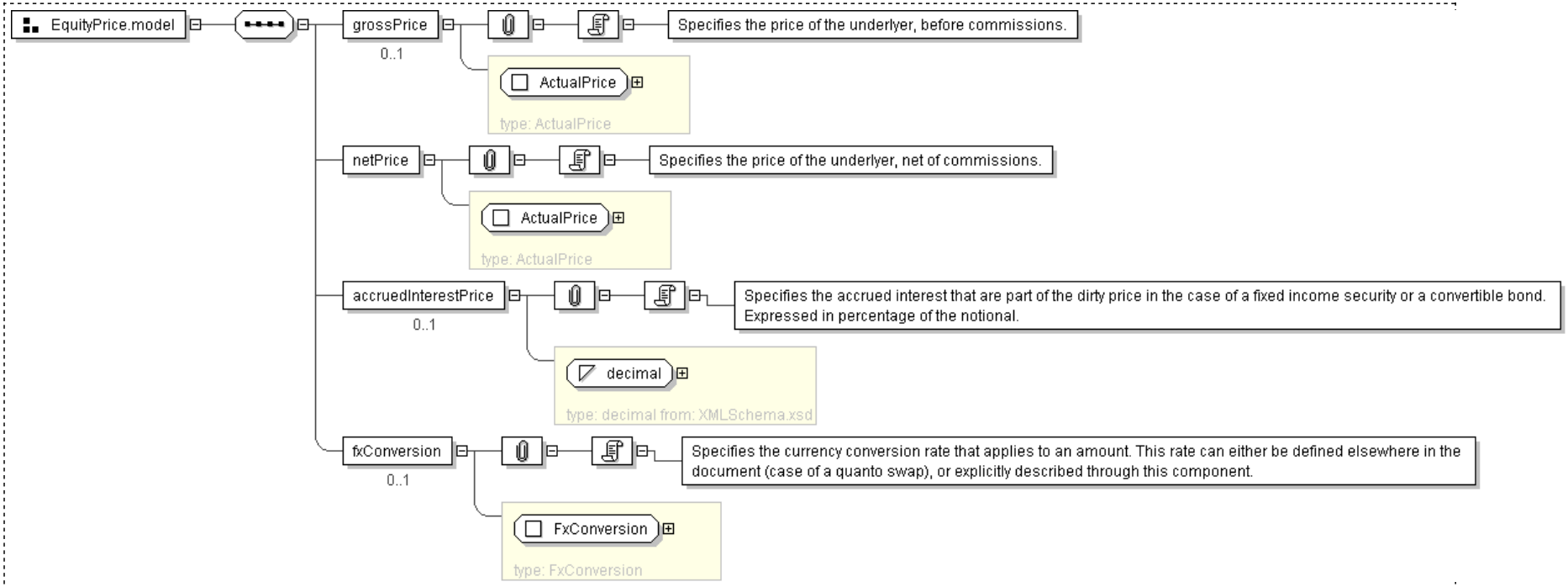
[Table of contents]

Name	EquityPrice.model
Used by (from the same schema document)	Complex Type Price , Complex Type Price

XML Instance Representation

```
<grossPrice> ActualPrice </grossPrice> [0..1]
'Specifies the price of the underlyer, before commissions.'

<netPrice> ActualPrice </netPrice> [1]
'Specifies the price of the underlyer, net of commissions.'

<accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
'Specifies the accrued interest that are part of the dirty price in the case of a fixed income security or a convertible bond. Expressed in
percentage of the notional.'

<fxConversion> FxConversion </fxConversion> [0..1]
'Specifies the currency conversion rate that applies to an amount. This rate can either be defined elsewhere in the document (case of a
quanto swap), or explicitly described through this component.'
```

Diagram



Schema Component Representation

```
<xsd:group name="EquityPrice.model">
  <xsd:sequence>
    <xsd:element name="grossPrice" type=" ActualPrice " minOccurs="0"/>
    <xsd:element name="netPrice" type=" ActualPrice "/>
    <xsd:element name="accruedInterestPrice" type=" xsd:decimal " minOccurs="0"/>
```

```
    <xsd:element name="fxConversion" type=" FxConversion " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: ExchangeIdentifier.model

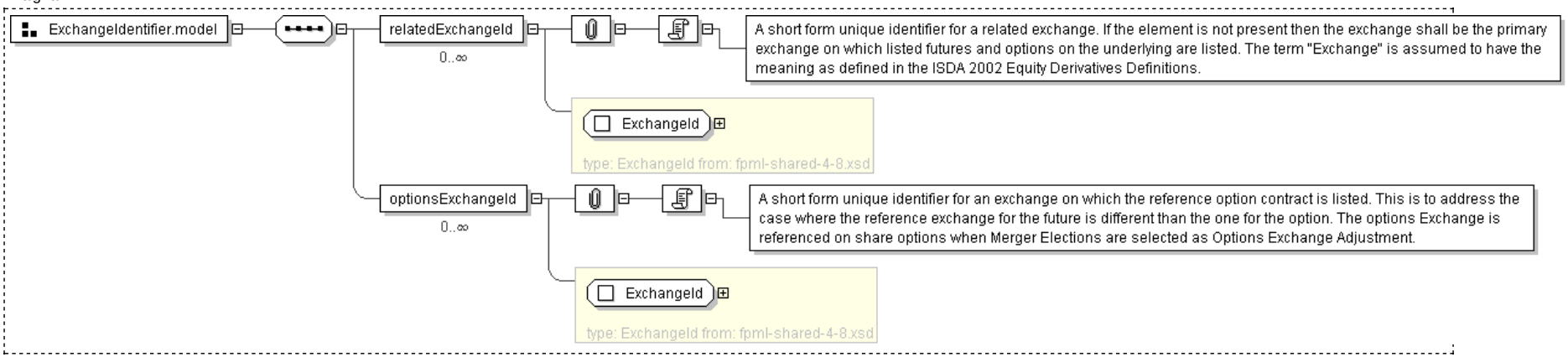
[Table of contents]

Name	ExchangeIdentifier.model
Used by (from the same schema document)	Complex Type ExchangeTraded

XML Instance Representation

```
<relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
'A short form unique identifier for a related exchange. If the element is not present then the exchange shall be the primary exchange on which listed futures and options on the underlying are listed. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

<optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
'A short form unique identifier for an exchange on which the reference option contract is listed. This is to address the case where the reference exchange for the future is different than the one for the option. The options Exchange is referenced on share options when Merger Elections are selected as Options Exchange Adjustment.'
```

Diagram



Schema Component Representation

```
<xsd:group name="ExchangeIdentifier.model">
  <xsd:sequence>
    <xsd:element name="relatedExchangeId" type=" ExchangeId " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="optionsExchangeId" type=" ExchangeId " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Quotation.model

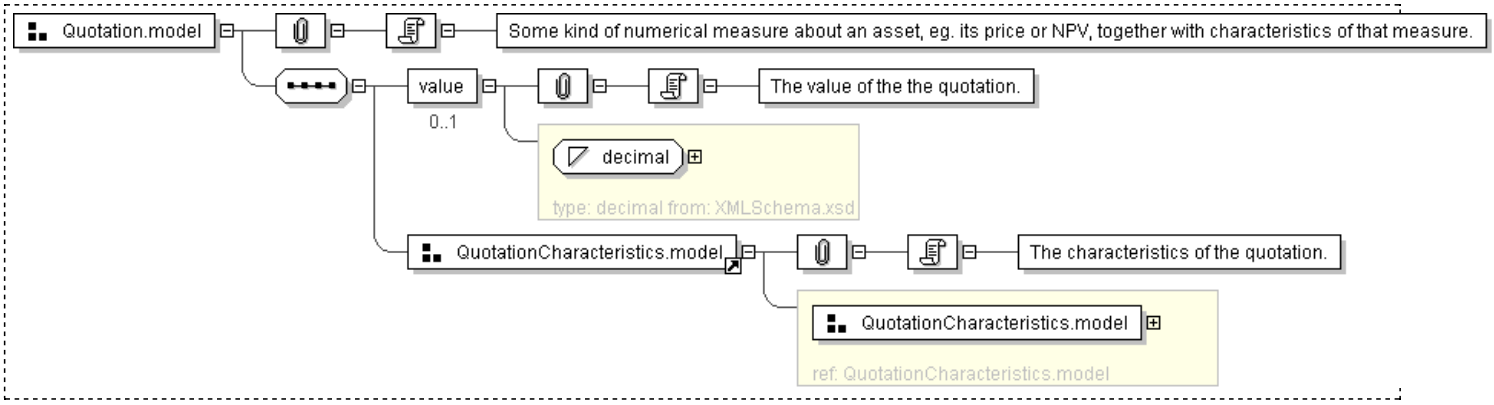
[Table of contents]

Name	Quotation.model
Used by (from the same schema document)	Complex Type BasicQuotation
Documentation	Some kind of numerical measure about an asset, eg. its price or NPV, together with characteristics of that measure.

XML Instance Representation

```
<value> xsd:decimal </value> [0..1]
'The value of the the quotation.'

<measureType> AssetMeasureType </measureType> [0..1]
'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

<quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
'The optional units that the measure is expressed in. If not supplied, this is assumed to be a price/value in
currency units.'

<side> QuotationSideEnum </side> [0..1]
'The side (bid/mid/ask) of the measure.'

<currency> Currency </currency> [0..1]
'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the
reportingCurrency in the valuationScenarioDefinition.'

<timing> QuoteTiming </timing> [0..1]
'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs also to
be supplied.'

Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


   Start Choice [1]
      <businessCenter> BusinessCenter </businessCenter> [1]
      'A city or other business center.'

      <exchangeId> ExchangeId </exchangeId> [1]
      'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

   End Choice
End Group: QuoteLocation.model
<informationSource> InformationSource </informationSource> [0..*]
'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page
3750.'

<time> xsd:dateTime </time> [0..1]
'When the quote was observed or derived.'

<valuationDate> xsd:date </valuationDate> [0..1]
'When the quote was computed.'

<expiryTime> xsd:dateTime </expiryTime> [0..1]
'When does the quote cease to be valid.'

<cashFlowType> CashflowType </cashFlowType> [0..1]
'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement Fee,
Brokerage Fee, etc.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Quotation.model">
   <xsd:sequence>
      <xsd:element name="value" type=" xsd:decimal " minOccurs="0"/>
      <xsd:group ref=" QuotationCharacteristics.model "/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: QuotationCharacteristics.model

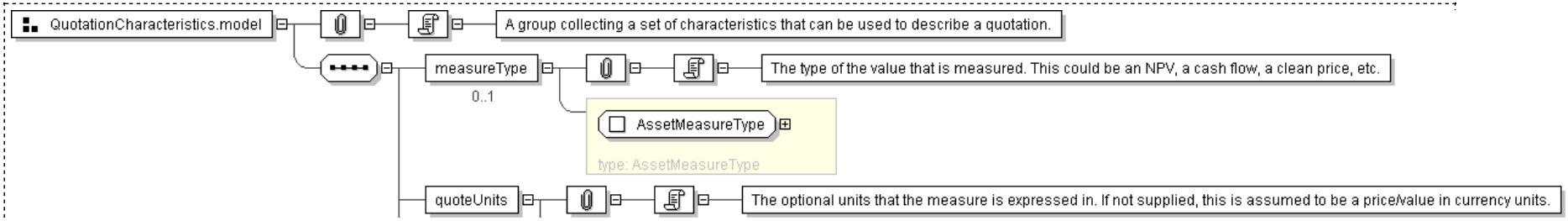
[Table of contents]

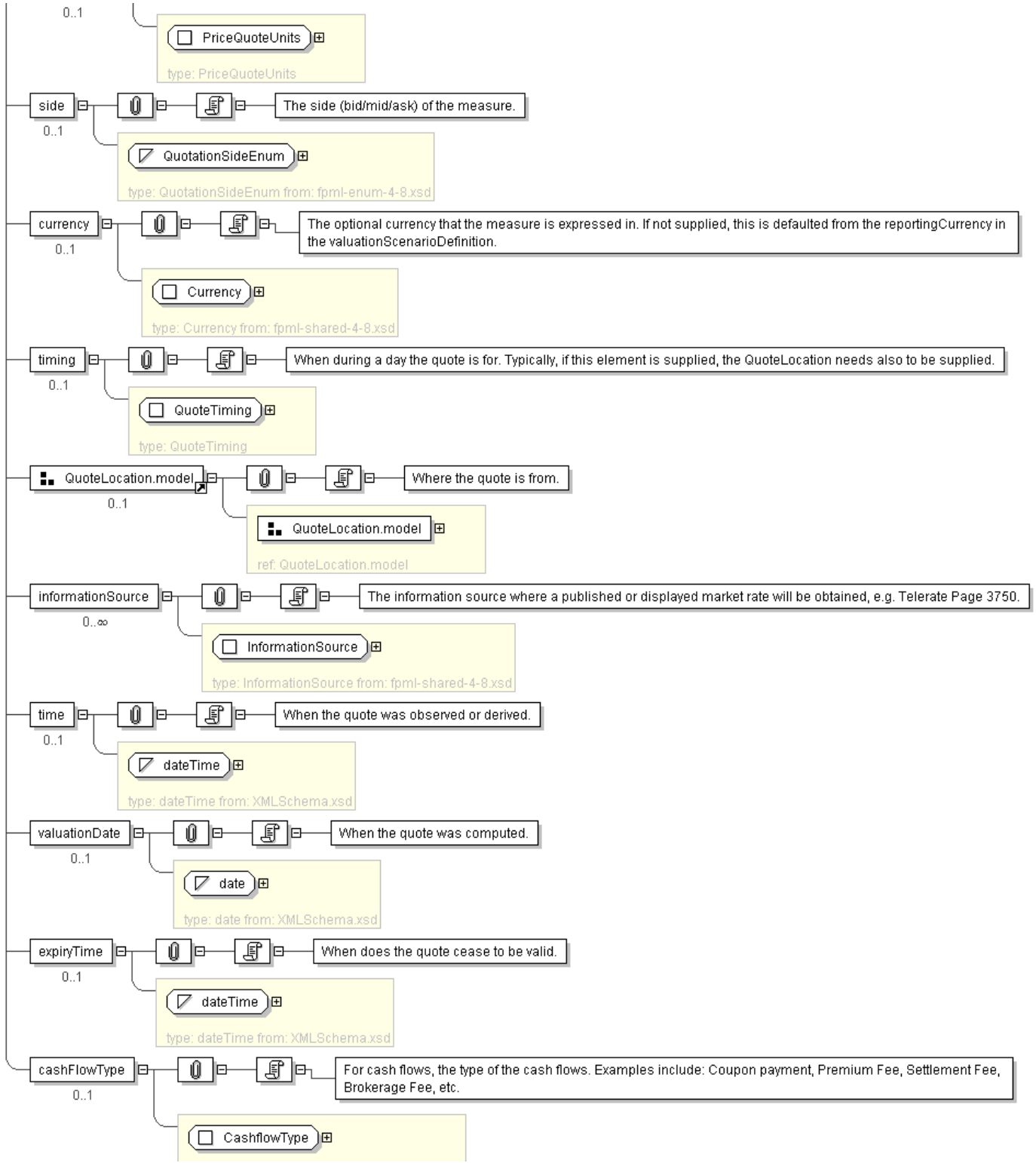
Name	QuotationCharacteristics.model
Used by (from the same schema document)	Complex Type QuotationCharacteristics , Model Group Quotation.model
Documentation	A group collecting a set of characteristics that can be used to describe a quotation.

XML Instance Representation

```
<measureType> AssetMeasureType </measureType> [0..1]
'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

<quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
'The optional units that the measure is expressed in. If not supplied, this is assumed to be a price/value in currency units.'

<side> QuotationSideEnum </side> [0..1]
'The side (bid/mid/ask) of the measure.'

<currency> Currency </currency> [0..1]
'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the reportingCurrency in the
valuationScenarioDefinition.'

<timing> QuoteTiming </timing> [0..1]
'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs also to be supplied.'

Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


  Start Choice [1]
    <businessCenter> BusinessCenter </businessCenter> [1]
    'A city or other business center.'

    <exchangeId> ExchangeId </exchangeId> [1]
    'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

  End Choice
End Group: QuoteLocation.model
<informationSource> InformationSource </informationSource> [0..*]
'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

<time> xsd:dateTime </time> [0..1]
'When the quote was observed or derived.'

<valuationDate> xsd:date </valuationDate> [0..1]
'When the quote was computed.'

<expiryTime> xsd:dateTime </expiryTime> [0..1]
'When does the quote cease to be valid.'

<cashFlowType> CashflowType </cashFlowType> [0..1]
'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement Fee, Brokerage Fee, etc.'
```

Diagram




0..1
PriceQuoteUnits
type: PriceQuoteUnits
side
0..1
The side (bid/mid/ask) of the measure.
QuotationSideEnum
type: QuotationSideEnum from: fpml-enum-4-8.xsd
currency
0..1
The optional currency that the measure is expressed in. If not supplied, this is defaulted from the reportingCurrency in the valuationScenarioDefinition.
Currency
type: Currency from: fpml-shared-4-8.xsd
timing
0..1
When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs also to be supplied.
QuoteTiming
type: QuoteTiming
QuoteLocation.model
0..1
Where the quote is from.
QuoteLocation.model
ref: QuoteLocation.model
informationSource
0..∞
The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.
InformationSource
type: InformationSource from: fpml-shared-4-8.xsd
time
0..1
When the quote was observed or derived.
dateTime
type: dateTime from: XMLSchema.xsd
valuationDate
0..1
When the quote was computed.
date
type: date from: XMLSchema.xsd
expiryTime
0..1
When does the quote cease to be valid.
dateTime
type: dateTime from: XMLSchema.xsd
cashFlowType
0..1
For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement Fee, Brokerage Fee, etc.
CashflowType

type: CashflowType from: fpml-shared-4-8.xsd

Schema Component Representation

```
<xsd:group name="QuotationCharacteristics.model">
  <xsd:sequence>
    <xsd:element name="measureType" type=" AssetMeasureType " minOccurs="0"/>
    <xsd:element name="quoteUnits" type=" PriceQuoteUnits " minOccurs="0"/>
    <xsd:element name="side" type=" QuotationSideEnum " minOccurs="0"/>
    <xsd:element name="currency" type=" Currency " minOccurs="0"/>
    <xsd:element name="timing" type=" QuoteTiming " minOccurs="0"/>
    <xsd:group ref=" QuoteLocation.model " minOccurs="0"/>
    <xsd:element name="informationSource" type=" InformationSource " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="time" type=" xsd:dateTime " minOccurs="0"/>
    <xsd:element name="valuationDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="expiryTime" type=" xsd:dateTime " minOccurs="0"/>
    <xsd:element name="cashFlowType" type=" CashflowType " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: QuoteLocation.model

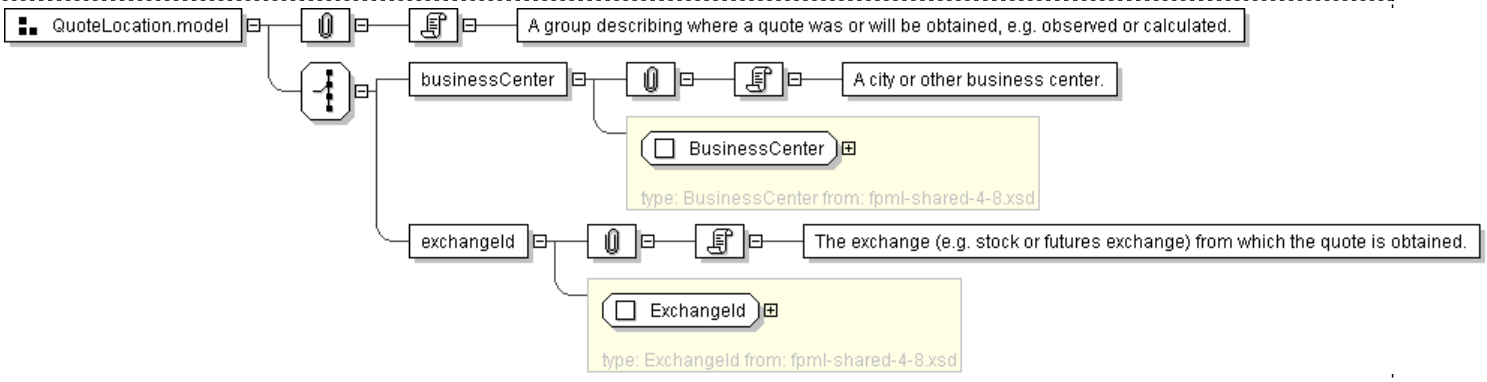
[Table of contents]

Name	QuoteLocation.model
Used by (from the same schema document)	Model Group QuotationCharacteristics.model
Documentation	A group describing where a quote was or will be obtained, e.g. observed or calculated.

XML Instance Representation

```
Start Choice [1]
   <businessCenter> BusinessCenter </businessCenter> [1]
   'A city or other business center.'

   <exchangeId> ExchangeId </exchangeId> [1]
   'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="QuoteLocation.model">
  <xsd:choice>
    <xsd:element name="businessCenter" type=" BusinessCenter "/>
    <xsd:element name="exchangeId" type=" ExchangeId "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ActualPrice

[Table of contents]

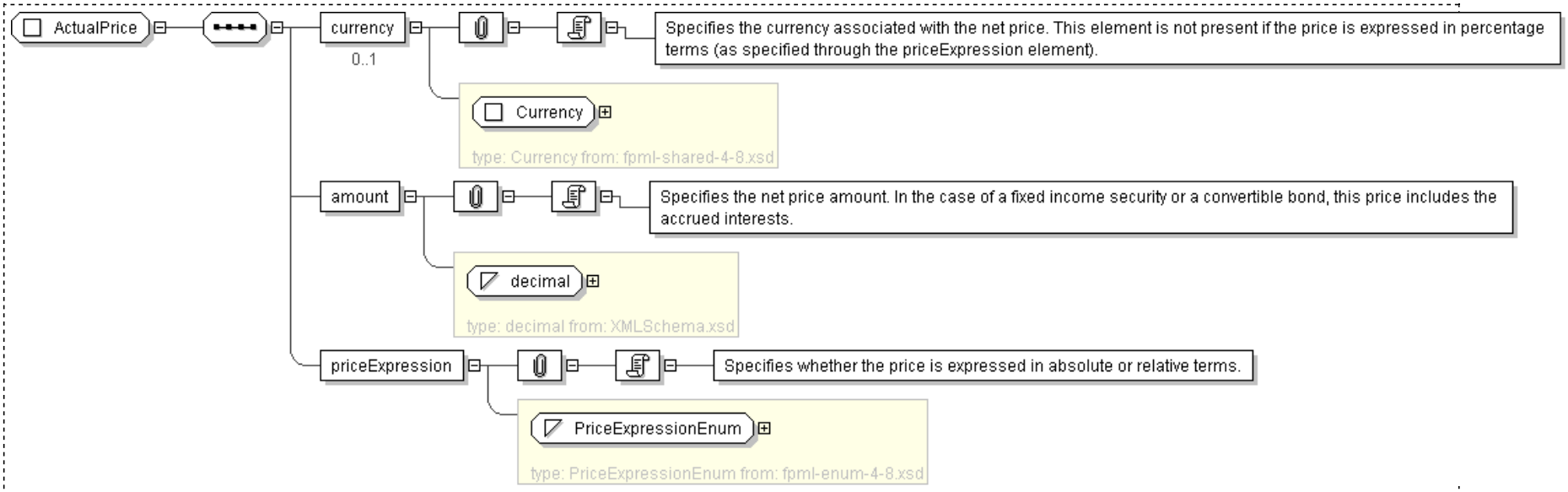
Super-types:	None
Sub-types:	None

Name	ActualPrice
Used by (from the same schema document)	Model Group EquityPrice.model , Model Group EquityPrice.model
Abstract	no

XML Instance Representation

```
<...>
   <currency> Currency </currency> [0..1]
   'Specifies the currency associated with the net price. This element is not present if the price is expressed in percentage
   terms (as specified through the priceExpression element).'

   <amount> xsd:decimal </amount> [1]
   'Specifies the net price amount. In the case of a fixed income security or a convertible bond, this price includes the accrued
   interests.'

   <priceExpression> PriceExpressionEnum </priceExpression> [1]
   'Specifies whether the price is expressed in absolute or relative terms.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ActualPrice">
   <xsd:sequence>
      <xsd:element name="currency" type=" Currency " minOccurs="0"/>
      <xsd:element name="amount" type=" xsd:decimal "/>
      <xsd:element name="priceExpression" type=" PriceExpressionEnum "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AnyAssetReference

[Table of contents]

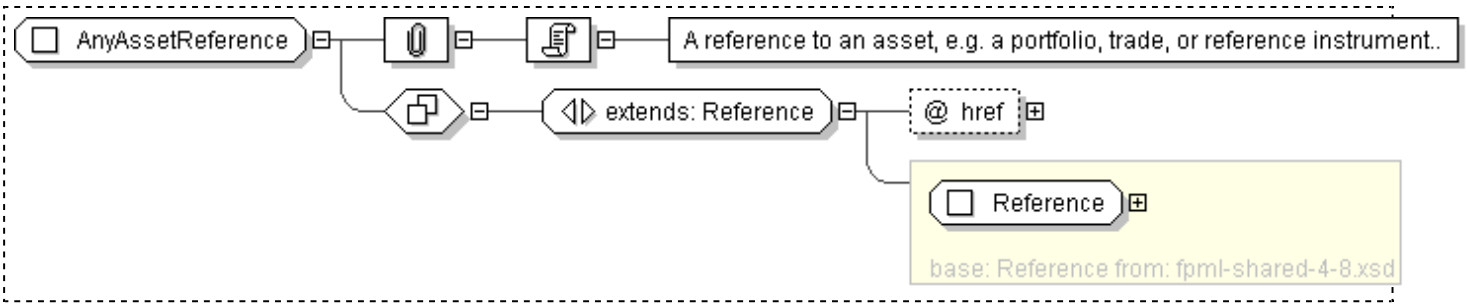
Super-types:	Reference < AnyAssetReference (by extension)
Sub-types:	None

Name	AnyAssetReference
Abstract	no
Documentation	A reference to an asset, e.g. a portfolio, trade, or reference instrument..

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AnyAssetReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Asset

[Table of contents]

Super-types: None

Sub-types:

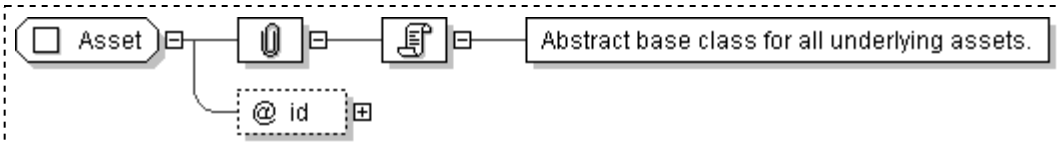
- Basket (by extension)
- IdentifiedAsset (by extension)
  - Cash (by extension)
  - Commodity (by extension)
  - UnderlyingAsset (by extension)
    - Deposit (by extension)
    - ExchangeTraded (by extension)
      - Bond (by extension)
        - ConvertibleBond (by extension)
      - EquityAsset (by extension)
      - ExchangeTradedCalculatedPrice (by extension)
        - ExchangeTradedFund (by extension)
        - Index (by extension)
      - ExchangeTradedContract (by extension)
      - Future (by extension)
    - FxRateAsset (by extension)
    - Loan (by extension)
    - Mortgage (by extension)
    - MutualFund (by extension)
    - RateIndex (by extension)
    - SimpleCreditDefaultSwap (by extension)
    - SimpleFra (by extension)
    - SimpleIRSwap (by extension)

Name	Asset
Used by (from the same schema document)	Element underlyingAsset
Abstract	yes
Documentation	Abstract base class for all underlying assets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Asset" abstract="true">
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssetMeasureType

[Table of contents]

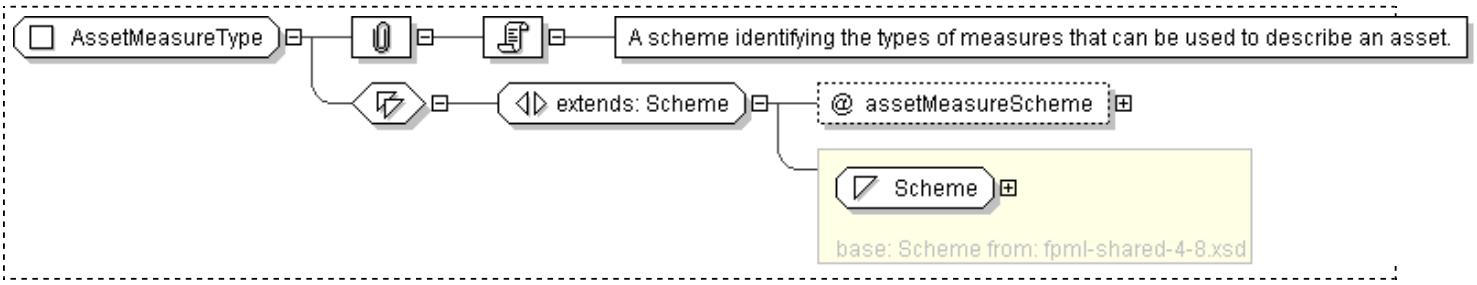
Super-types:	Scheme < AssetMeasureType (by extension)
Sub-types:	None

Name	AssetMeasureType
Used by (from the same schema document)	Model Group QuotationCharacteristics.model
Abstract	no
Documentation	A scheme identifying the types of measures that can be used to describe an asset.

XML Instance Representation

```
<...
 assetMeasureScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssetMeasureType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="assetMeasureScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/asset-measure"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssetPool

[Table of contents]

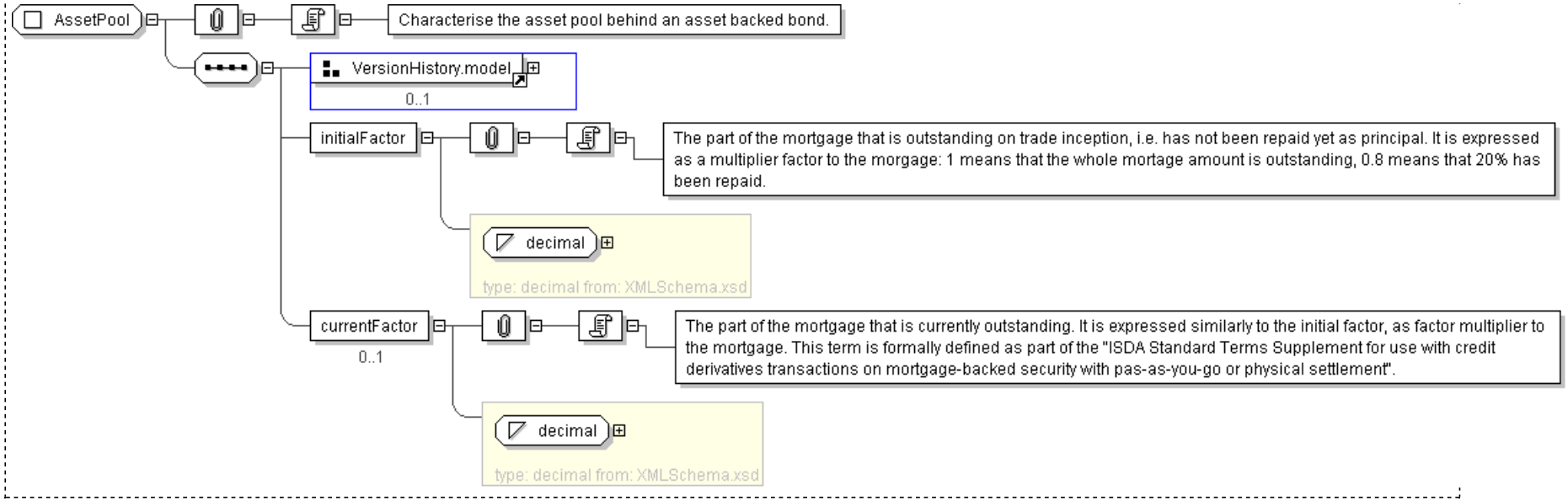
Super-types:	None
Sub-types:	None

Name	AssetPool
Used by (from the same schema document)	Complex Type Mortgage
Abstract	no
Documentation	Characterise the asset pool behind an asset backed bond.

XML Instance Representation

```
<...>
   Start Group: VersionHistory.model [0..1]
      <version> xsd:nonNegativeInteger </version> [1]
      'The version number'

      <effectiveDate> IdentifiedDate </effectiveDate> [0..1]
      'Optionally it is possible to specify a version effective date when a versionId is supplied.'

   End Group: VersionHistory.model
   <initialFactor> xsd:decimal </initialFactor> [1]
   'The part of the mortgage that is outstanding on trade inception, i.e. has not been repaid yet as principal. It is expressed as
   a multiplier factor to the morgage: 1 means that the whole mortage amount is outstanding, 0.8 means that 20% has been repaid.'

   <currentFactor> xsd:decimal </currentFactor> [0..1]
   'The part of the mortgage that is currently outstanding. It is expressed similarly to the initial factor, as factor multiplier
   to the mortgage. This term is formally defined as part of the \"ISDA Standard Terms Supplement for use with credit derivatives
   transactions on mortgage-backed security with pas-as-you-go or physical settlement\".'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssetPool">
  <xsd:sequence>
    <xsd:group ref=" VersionHistory.model " minOccurs="0"/>
    <xsd:element name="initialFactor" type=" xsd:decimal "/>
    <xsd:element name="currentFactor" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssetReference

[Table of contents]

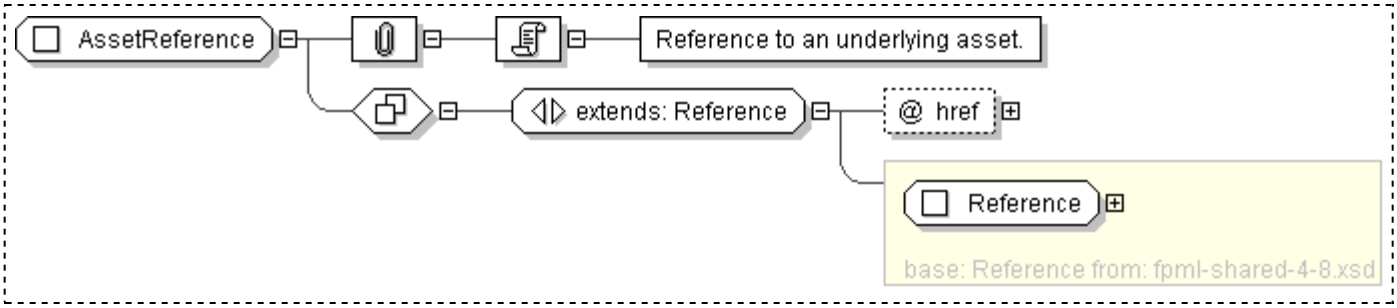
Super-types:	Reference < AssetReference (by extension)
Sub-types:	None

Name	AssetReference
Abstract	no
Documentation	Reference to an underlying asset.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssetReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Asset"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasicQuotation

[Table of contents]

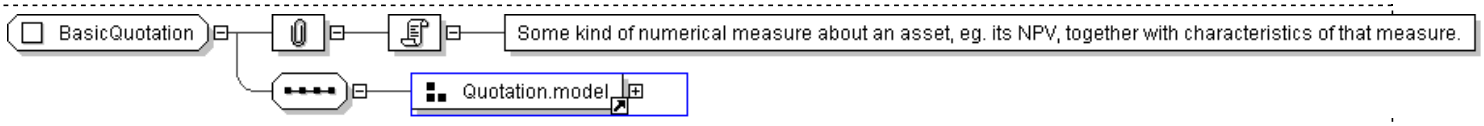
Super-types:	None
Sub-types:	None

Name	BasicQuotation
Abstract	no
Documentation	Some kind of numerical measure about an asset, eg. its NPV, together with characteristics of that measure.

XML Instance Representation

```
<...>
   <value> xsd:decimal </value> [0..1]
   'The value of the the quotation.'

   <measureType> AssetMeasureType </measureType> [0..1]
   'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

   <quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
   'The optional units that the measure is expressed in. If not supplied, this is assumed to be a
   price/value in currency units.'

   <side> QuotationSideEnum </side> [0..1]
   'The side (bid/mid/ask) of the measure.'

   <currency> Currency </currency> [0..1]
   'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the
   reportingCurrency in the valuationScenarioDefinition.'

   <timing> QuoteTiming </timing> [0..1]
   'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs
   also to be supplied.'

   Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


      Start Choice [1]
         <businessCenter> BusinessCenter </businessCenter> [1]
         'A city or other business center.'

         <exchangeId> ExchangeId </exchangeId> [1]
         'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

      End Choice
   End Group: QuoteLocation.model
   <informationSource> InformationSource </informationSource> [0..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate
   Page 3750.'

   <time> xsd:dateTime </time> [0..1]
   'When the quote was observed or derived.'

   <valuationDate> xsd:date </valuationDate> [0..1]
   'When the quote was computed.'

   <expiryTime> xsd:dateTime </expiryTime> [0..1]
   'When does the quote cease to be valid.'

   <cashFlowType> CashflowType </cashFlowType> [0..1]
   'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee,
   Settlement Fee, Brokerage Fee, etc.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasicQuotation">
  <xsd:sequence>
    <xsd:group ref=" Quotation.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Basket

[Table of contents]

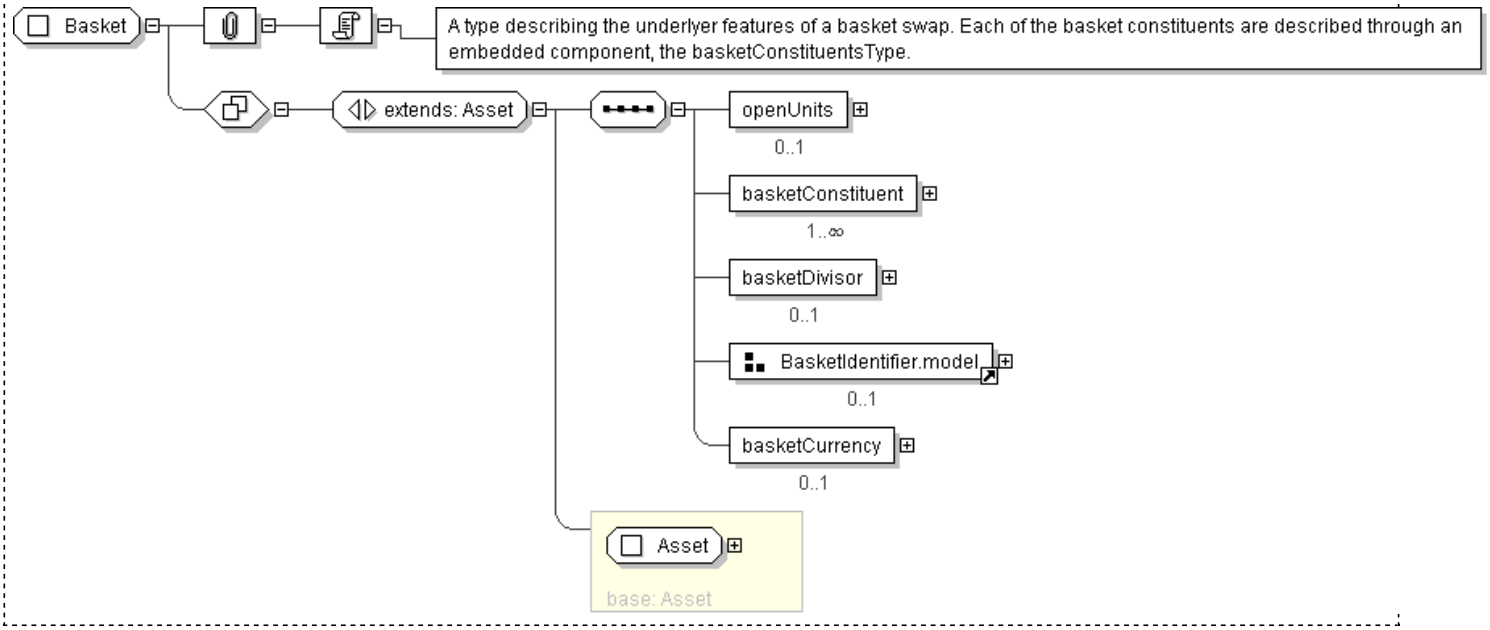
Super-types:	Asset < Basket (by extension)
Sub-types:	None

Name	Basket
Used by (from the same schema document)	Complex Type Underlyer , Element basket
Abstract	no
Documentation	A type describing the underlyer features of a basket swap. Each of the basket constituents are described through an embedded component, the basketConstituentsType.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <openUnits> xsd:decimal </openUnits> [0..1]
    'The number of units (index or securities) that constitute the underlyer of the swap. In the case of
    a basket swap, this element is used to reference both the number of basket units, and the number of
    each asset components of the basket when these are expressed in absolute terms.'

   <basketConstituent> BasketConstituent </basketConstituent> [1..*]
    'Describes each of the components of the basket.'

   <basketDivisor> xsd:decimal </basketDivisor> [0..1]
    'Specifies the basket divisor amount. This value is normally used to adjust the constituent weight
    for pricing or to adjust for dividends, or other corporate actions.'

   Start Group: BasketIdentifier.model [0..1]
 'Reuses the group that specifies a name and an identifier for a given basket.'


      Start Choice [1]
         <basketName> BasketName </basketName> [1]
          'The name of the basket expressed as a free format string. FpML does not define usage rules for
          this element.'

         <basketId> BasketId </basketId> [0..*]
          'A CDS basket identifier'

         <basketId> BasketId </basketId> [1..*]
          'A CDS basket identifier'

      End Choice
   End Group: BasketIdentifier.model
   <basketCurrency> Currency </basketCurrency> [0..1]
    'Specifies the currency for this basket.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Basket">
  <xsd:complexContent>
    <xsd:extension base=" Asset ">
      <xsd:sequence>
        <xsd:element name="openUnits" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="basketConstituent" type=" BasketConstituent " maxOccurs="unbounded"/>
        <xsd:element name="basketDivisor" type=" xsd:decimal " minOccurs="0"/>
        <xsd:group ref=" BasketIdentifier.model " minOccurs="0"/>
        <xsd:element name="basketCurrency" type=" Currency " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasketConstituent

[Table of contents]

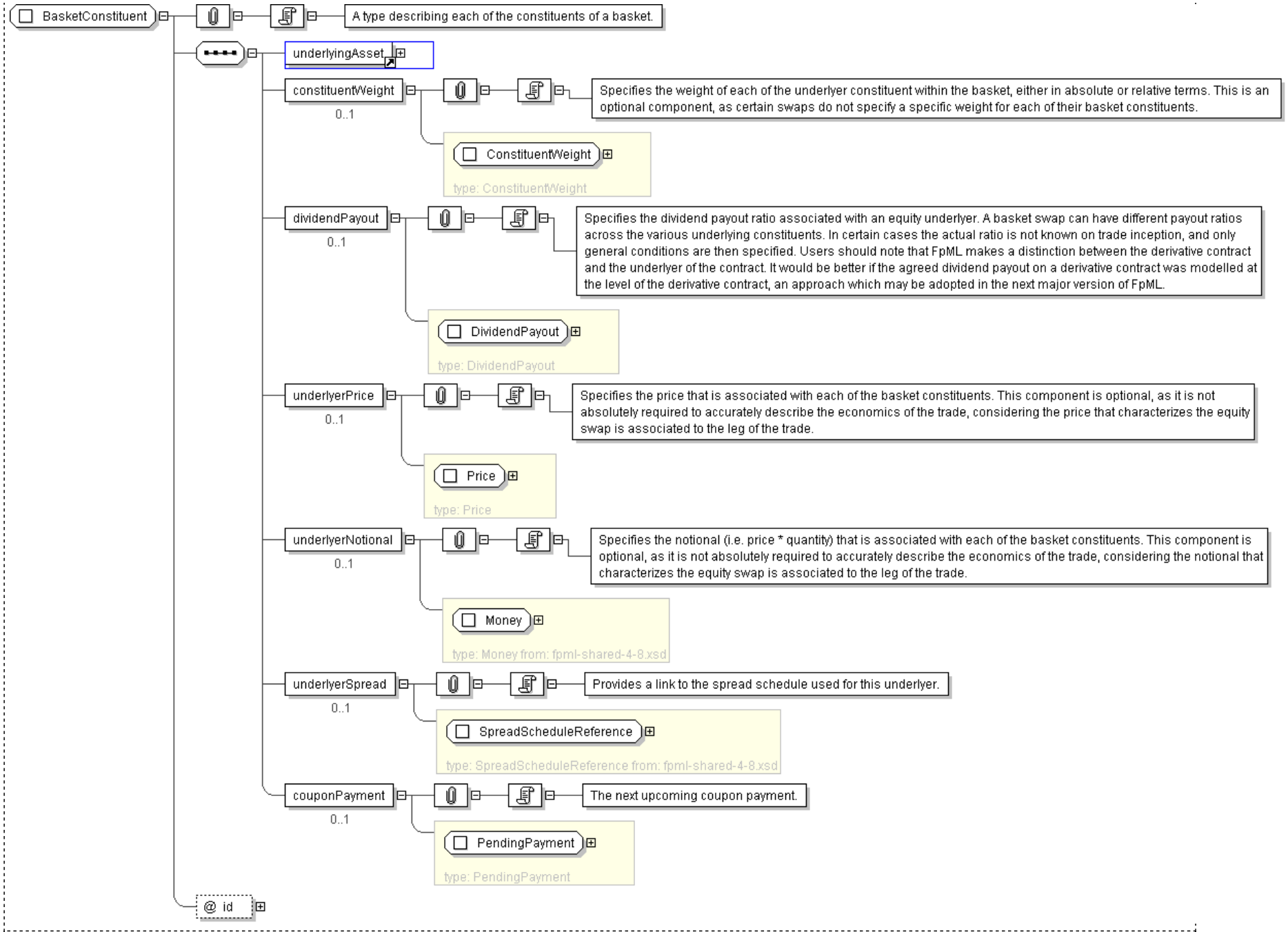
Super-types:	None
Sub-types:	None

Name	BasketConstituent
Used by (from the same schema document)	Complex Type Basket
Abstract	no
Documentation	A type describing each of the constituents of a basket.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <underlyingAsset> ... </underlyingAsset> [1]
   <constituentWeight> ConstituentWeight </constituentWeight> [0..1]
   'Specifies the weight of each of the underlyer constituent within the basket, either in absolute or relative terms. This is an optional
   component, as certain swaps do not specify a specific weight for each of their basket constituents.'

   <dividendPayout> DividendPayout </dividendPayout> [0..1]
   'Specifies the dividend payout ratio associated with an equity underlyer. A basket swap can have different payout ratios across the
   various underlying constituents. In certain cases the actual ratio is not known on trade inception, and only general conditions are then
   specified. Users should note that FpML makes a distinction between the derivative contract and the underlyer of the contract. It would be
   better if the agreed dividend payout on a derivative contract was modelled at the level of the derivative contract, an approach which may
   be adopted in the next major version of FpML.'

   <underlyerPrice> Price </underlyerPrice> [0..1]
   'Specifies the price that is associated with each of the basket constituents. This component is optional, as it is not absolutely
   required to accurately describe the economics of the trade, considering the price that characterizes the equity swap is associated to the
   leg of the trade.'

   <underlyerNotional> Money </underlyerNotional> [0..1]
   'Specifies the notional (i.e. price * quantity) that is associated with each of the basket constituents. This component is optional, as
   it is not absolutely required to accurately describe the economics of the trade, considering the notional that characterizes the equity
   swap is associated to the leg of the trade.'

   <underlyerSpread> SpreadScheduleReference </underlyerSpread> [0..1]
   'Provides a link to the spread schedule used for this underlyer.'

   <couponPayment> PendingPayment </couponPayment> [0..1]
   'The next upcoming coupon payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasketConstituent">
  <xsd:sequence>
    <xsd:element ref=" underlyingAsset "/>
    <xsd:element name="constituentWeight" type=" ConstituentWeight " minOccurs="0"/>
    <xsd:element name="dividendPayout" type=" DividendPayout " minOccurs="0"/>
    <xsd:element name="underlyerPrice" type=" Price " minOccurs="0"/>
```

```
      <xsd:element name="underlyerNotional" type=" Money " minOccurs="0"/>
      <xsd:element name="underlyerSpread" type=" SpreadScheduleReference " minOccurs="0"/>
      <xsd:element name="couponPayment" type=" PendingPayment " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasketId

[Table of contents]

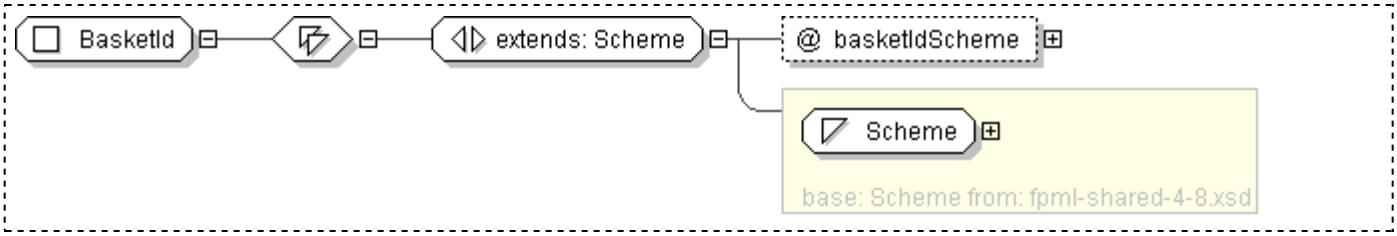
Super-types:	Scheme < BasketId (by extension)
Sub-types:	None

Name	BasketId
Used by (from the same schema document)	Model Group BasketIdentifier.model , Model Group BasketIdentifier.model
Abstract	no

XML Instance Representation

```
<...
 basketIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasketId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="basketIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasketName

[Table of contents]

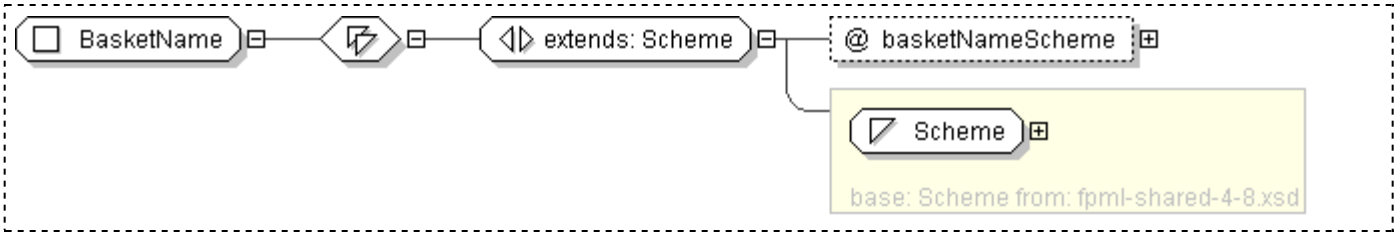
Super-types:	Scheme < BasketName (by extension)
Sub-types:	None

Name	BasketName
Used by (from the same schema document)	Model Group BasketIdentifier.model
Abstract	no

XML Instance Representation

```
<...
 basketNameScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasketName">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="basketNameScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Bond

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < Bond (by extension)
Sub-types:	• ConvertibleBond (by extension)

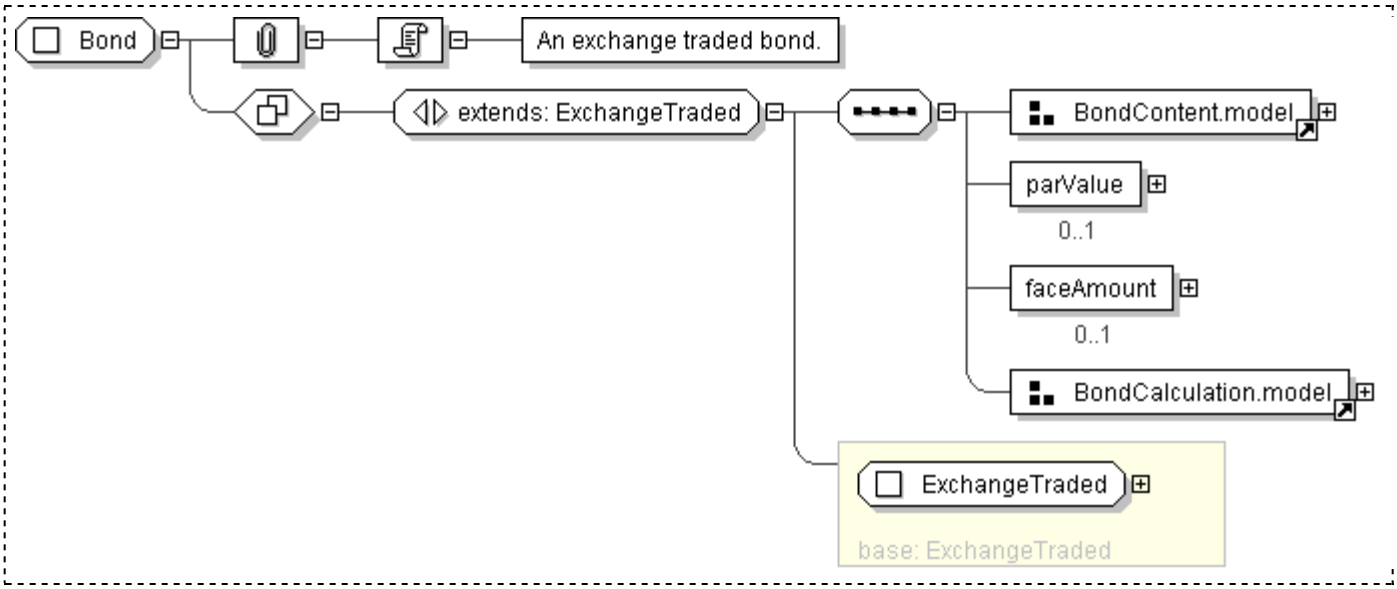
Name	Bond
Used by (from the same schema document)	Element bond
Abstract	no
Documentation	An exchange traded bond.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange
    for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'

   Start Choice [0..1]
 'Specifies the issuer name of a fixed income security or convertible bond. This
 name can either be explicitly stated, or specified as an href into another element
```

```
of the document, such as the obligor'

      <issuerName> xsd:string </issuerName> [1]
      <issuerPartyReference> PartyReference </issuerPartyReference> [1]
   End Choice
   <seniority> CreditSeniority </seniority> [0..1]
   'The repayment precedence of a debt instrument.'

   <couponType> CouponType </couponType> [0..1]
   'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-
   coupon.'

   <couponRate> xsd:decimal </couponRate> [0..1]
   'Specifies the coupon rate (expressed in percentage) of a fixed income security
   or convertible bond.'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of a security becomes due and payable.'

   <parValue> xsd:decimal </parValue> [0..1]
   'Specifies the nominal amount of a fixed income security or convertible bond.'

   <faceAmount> xsd:decimal </faceAmount> [0..1]
   'Specifies the total amount of the issue. Corresponds to the par value
   multiplied by the number of issued security.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
   'Specifies the frequency at which the bond pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
   'The day count basis for the bond.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Bond">
   <xsd:complexContent>
      <xsd:extension base=" ExchangeTraded ">
         <xsd:sequence>
            <xsd:group ref=" BondContent.model "/>
            <xsd:element name="parValue" type=" xsd:decimal " minOccurs="0"/>
            <xsd:element name="faceAmount" type=" xsd:decimal " minOccurs="0"/>
            <xsd:group ref=" BondCalculation.model "/>
         </xsd:sequence>
      </xsd:extension>
```

```
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Cash

[Table of contents]

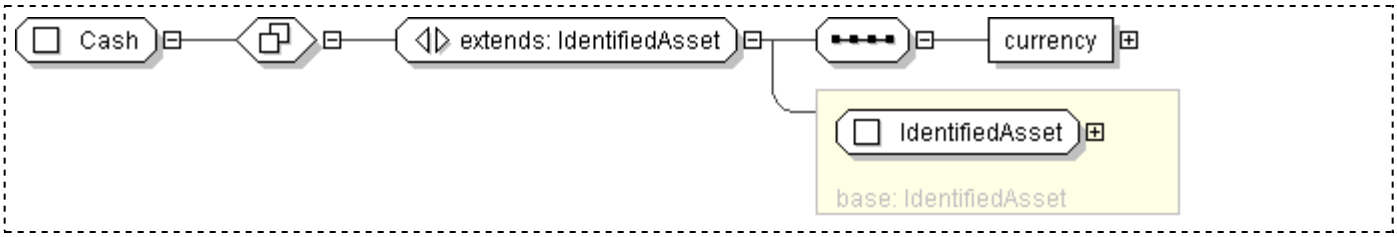
Super-types:	Asset < IdentifiedAsset (by extension) < Cash (by extension)
Sub-types:	None

Name	Cash
Used by (from the same schema document)	Element cash
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private
   identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Cash">
   <xsd:complexContent>
      <xsd:extension base=" IdentifiedAsset ">
         <xsd:sequence>
            <xsd:element name="currency" type=" Currency "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Commission

[Table of contents]

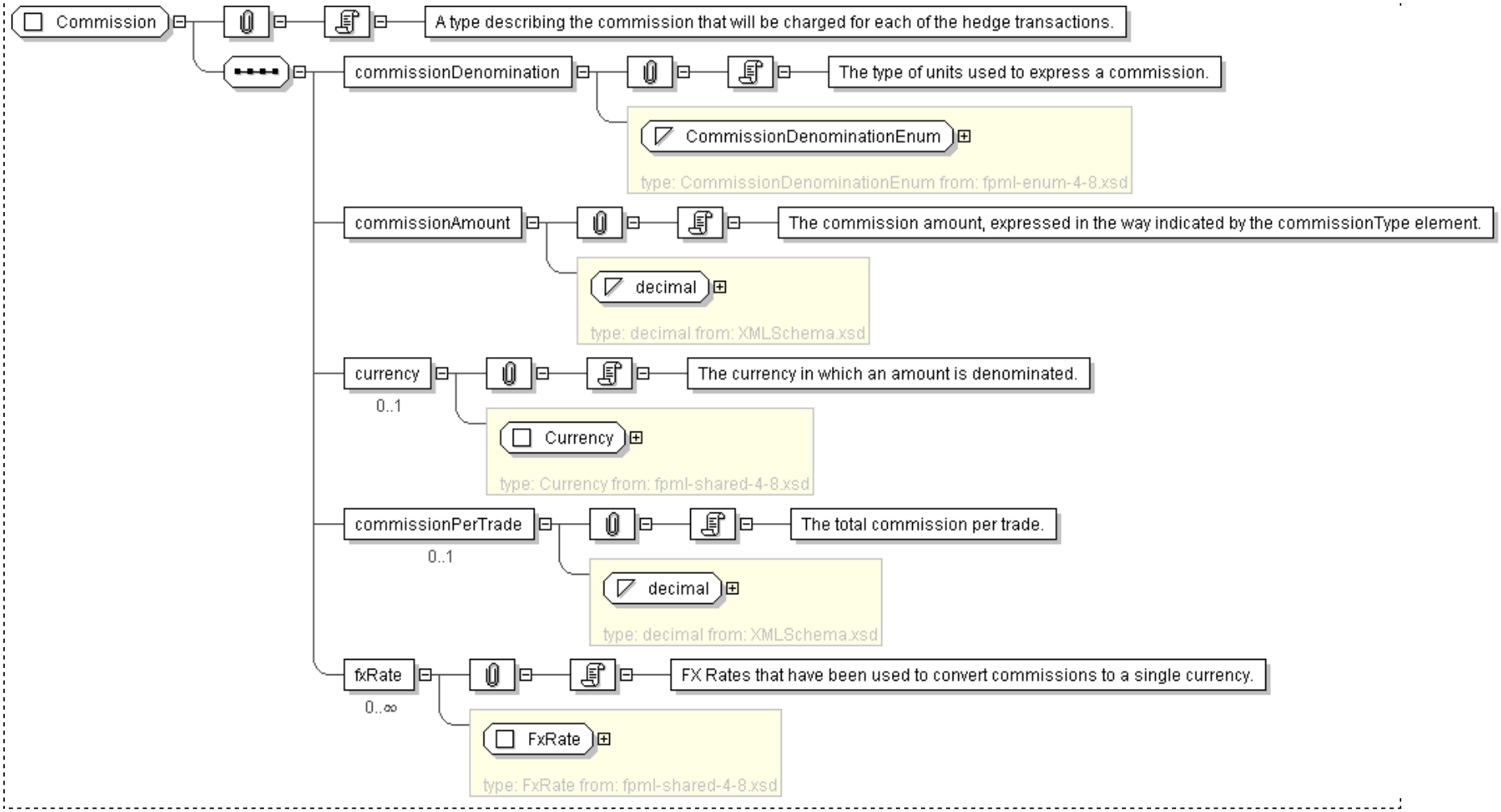
Super-types:	None
Sub-types:	None

Name	Commission
Used by (from the same schema document)	Complex Type Price
Abstract	no
Documentation	A type describing the commission that will be charged for each of the hedge transactions.

XML Instance Representation

```
<...>
   <commissionDenomination> CommissionDenominationEnum </commissionDenomination> [1]
   'The type of units used to express a commission.'

   <commissionAmount> xsd:decimal </commissionAmount> [1]
   'The commission amount, expressed in the way indicated by the commissionType element.'

   <currency> Currency </currency> [0..1]
   'The currency in which an amount is denominated.'

   <commissionPerTrade> xsd:decimal </commissionPerTrade> [0..1]
   'The total commission per trade.'

   <fxRate> FxRate </fxRate> [0..*]
   'FX Rates that have been used to convert commissions to a single currency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Commission">
  <xsd:sequence>
    <xsd:element name="commissionDenomination" type=" CommissionDenominationEnum "/>
    <xsd:element name="commissionAmount" type=" xsd:decimal "/>
    <xsd:element name="currency" type=" Currency " minOccurs="0"/>
    <xsd:element name="commissionPerTrade" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="fxRate" type=" FxRate " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Commodity

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < Commodity (by extension)
Sub-types:	None

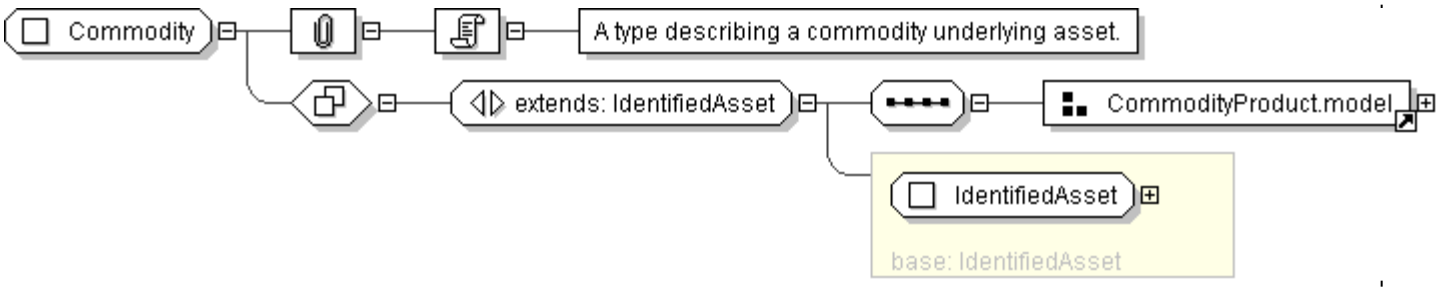
Name	Commodity
Used by (from the same schema document)	Element commodity
Abstract	no
Documentation	A type describing a commodity underlying asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   Start Group: CommodityReferencePriceFramework.model [0..1]
      <commodityBase> CommodityBase </commodityBase> [1]
       'A coding scheme value to identify the base type of the commodity being
       traded. Where possible, this should follow the naming convention used in the
       2005 ISDA Commodity Definitions. For example, \'Oil\'.'

      <commodityDetails> CommodityDetails </commodityDetails> [1]
       'A coding scheme value to identify the commodity being traded more
       specifically. Where possible, this should follow the naming convention used
       in the 2005 ISDA Commodity Definitions. For example, \'Brent\'.'

      <unit> QuantityUnit </unit> [1]
       'A coding scheme value to identify the unit in which the undelryer is
       denominated. Where possible, this should follow the naming convention used in
       the 2005 ISDA Commodity Definitions.'

      <currency> Currency </currency> [1]
       'The currency in which the Commodity Reference Price is published.'

      Start Choice [1]
         <exchangeId> ExchangeId </exchangeId> [1]
          'For those commodities being traded with reference to the price of a
          listed future, the exchange where that future is listed should be
          specified here.'

         <publication> InformationSource </publication> [1]
          'For those commodities being traded with reference to a price distributed
          by a publication, that publication should be specified here.'

      End Choice
   End Group: CommodityReferencePriceFramework.model
   <specifiedPrice> SpecifiedPriceEnum </specifiedPrice> [1]
    'The Specified Price is not defined in the Commodity Reference Price and so
    needs to be stated in the Underlyer definition as it will impact the
    calculation of the Floating Price.'
```

```
Start Sequence [0..1]
   Start Choice [1]
      <deliveryDates> DeliveryDatesEnum </deliveryDates> [1]
      'The Delivery Date is a NearbyMonth, for use when the Commodity
      Transaction references Futures Contract.'

      <deliveryDate> AdjustableDate </deliveryDate> [1]
      'The Delivery Date is a fixed, single day.'

      <deliveryDateYearMonth> xsd:gYearMonth </deliveryDateYearMonth> [1]
      'The Delivery Date is a fixed, single month.'

   End Choice
   <deliveryDateRollConvention> Offset </deliveryDateRollConvention> [0..1]
   'Specifies, for a Commodity Transaction that references a listed future via
   the deliveryDates element, the day on which the specified future will roll to
   the next nearby month when the referenced future expires. If the future will
   not roll at all - i.e. the price will be taken from the expiring contract, 0
   should be specified here. If the future will roll to the next nearby on the
   last trading day - i.e. the price will be taken from the next nearby on the
   last trading day, then 1 should be specified and so on.'

End Sequence
<multiplier> PositiveDecimal </multiplier> [0..1]
'Specifies the multiplier associated with a Transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Commodity">
   <xsd:complexContent>
      <xsd:extension base=" IdentifiedAsset ">
         <xsd:sequence>
            <xsd:group ref=" CommodityProduct.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityBase

[Table of contents]

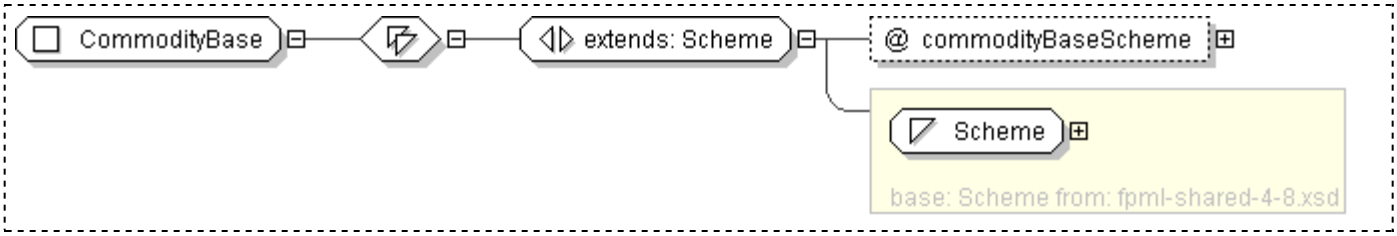
Super-types:	Scheme < CommodityBase (by extension)
Sub-types:	None

Name	CommodityBase
Used by (from the same schema document)	Model Group CommodityReferencePriceFramework.model
Abstract	no

XML Instance Representation

```
<...
 commodityBaseScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityBase">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityBaseScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityBusinessCalendar

[Table of contents]

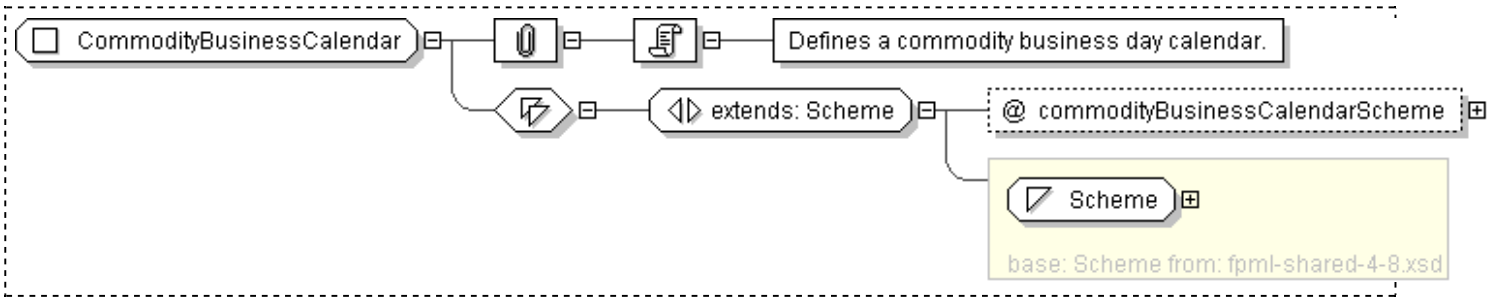
Super-types:	Scheme < CommodityBusinessCalendar (by extension)
Sub-types:	None

Name	CommodityBusinessCalendar
Used by (from the same schema document)	Complex Type CommodityBusinessCalendarTime
Abstract	no
Documentation	Defines a commodity business day calendar.

XML Instance Representation

```
<...
 commodityBusinessCalendarScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityBusinessCalendar">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityBusinessCalendarScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-business-calendar"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityBusinessCalendarTime

[Table of contents]

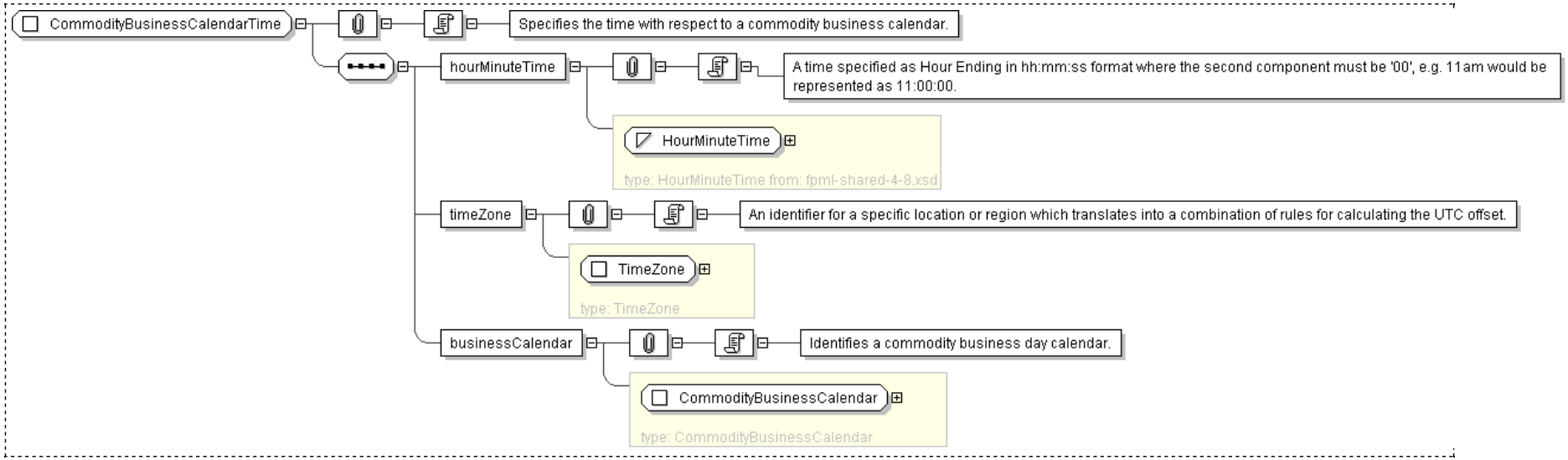
Super-types:	None
Sub-types:	None

Name	CommodityBusinessCalendarTime
Abstract	no
Documentation	Specifies the time with respect to a commodity business calendar.

XML Instance Representation

```
<...>
   <hourMinuteTime> HourMinuteTime </hourMinuteTime> [1]
    'A time specified as Hour Ending in hh:mm:ss format where the second component must be \'00\', e.g. 11am would be represented as 11:00:00.'

   <timeZone> TimeZone </timeZone> [1]
    'An identifier for a specific location or region which translates into a combination of rules for calculating the UTC offset.'

   <businessCalendar> CommodityBusinessCalendar </businessCalendar> [1]
    'Identifies a commodity business day calendar.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityBusinessCalendarTime">
  <xsd:sequence>
    <xsd:element name="hourMinuteTime" type=" HourMinuteTime "/>
    <xsd:element name="timeZone" type=" TimeZone "/>
    <xsd:element name="businessCalendar" type=" CommodityBusinessCalendar "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityDetails

[Table of contents]

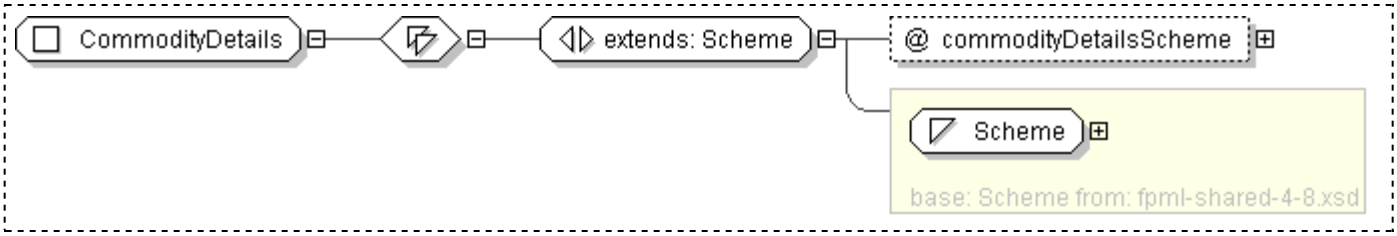
Super-types:	Scheme < CommodityDetails (by extension)
Sub-types:	None

Name	CommodityDetails
Used by (from the same schema document)	Model Group CommodityReferencePriceFramework.model
Abstract	no

XML Instance Representation

```
<...
 commodityDetailsScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityDetails">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityDetailsScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ConstituentWeight

[Table of contents]

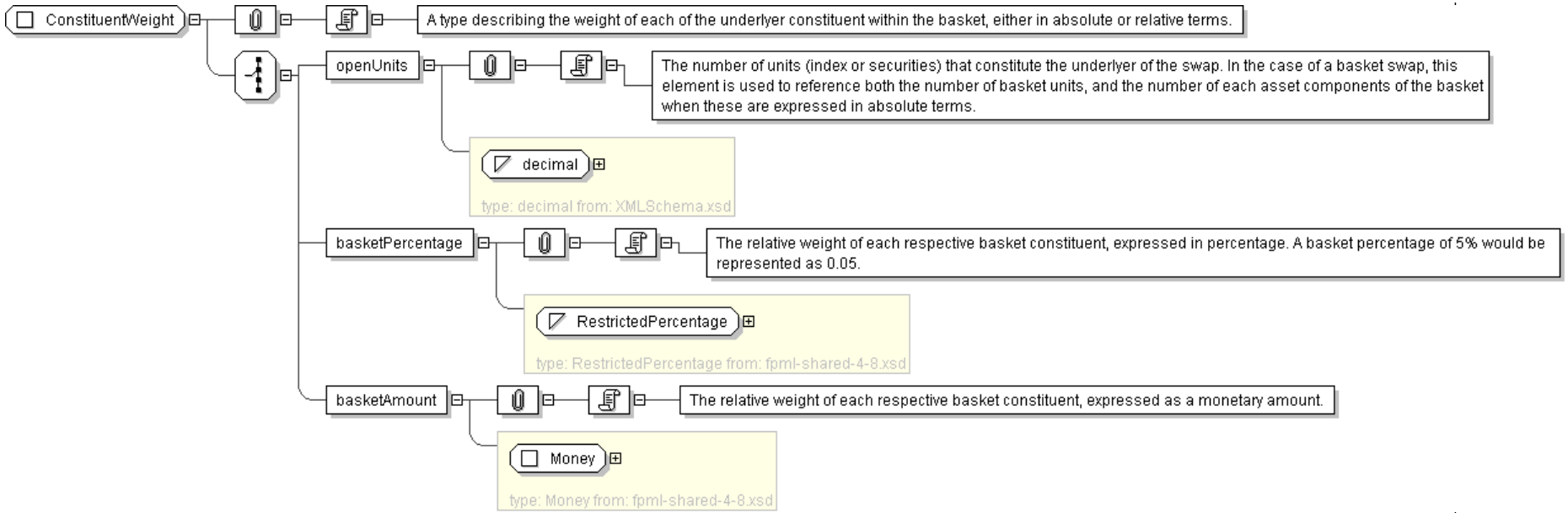
Super-types:	None
Sub-types:	None

Name	ConstituentWeight
Used by (from the same schema document)	Complex Type BasketConstituent
Abstract	no
Documentation	A type describing the weight of each of the underlyer constituent within the basket, either in absolute or relative terms.

XML Instance Representation

```
<...>
  Start Choice [1]
    <openUnits> xsd:decimal </openUnits> [1]
    'The number of units (index or securities) that constitute the underlyer of the swap. In the case of a basket swap, this element is
    used to reference both the number of basket units, and the number of each asset components of the basket when these are expressed in
    absolute terms.'

    <basketPercentage> RestrictedPercentage </basketPercentage> [1]
    'The relative weight of each respective basket constituent, expressed in percentage. A basket percentage of 5% would be represented as
    0.05.'

    <basketAmount> Money </basketAmount> [1]
    'The relative weight of each respective basket constituent, expressed as a monetary amount.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ConstituentWeight">
  <xsd:choice>
    <xsd:element name="openUnits" type=" xsd:decimal "/>
    <xsd:element name="basketPercentage" type=" RestrictedPercentage "/>
    <xsd:element name="basketAmount" type=" Money "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ConvertibleBond

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < Bond (by extension) < ConvertibleBond (by extension)
Sub-types:	None

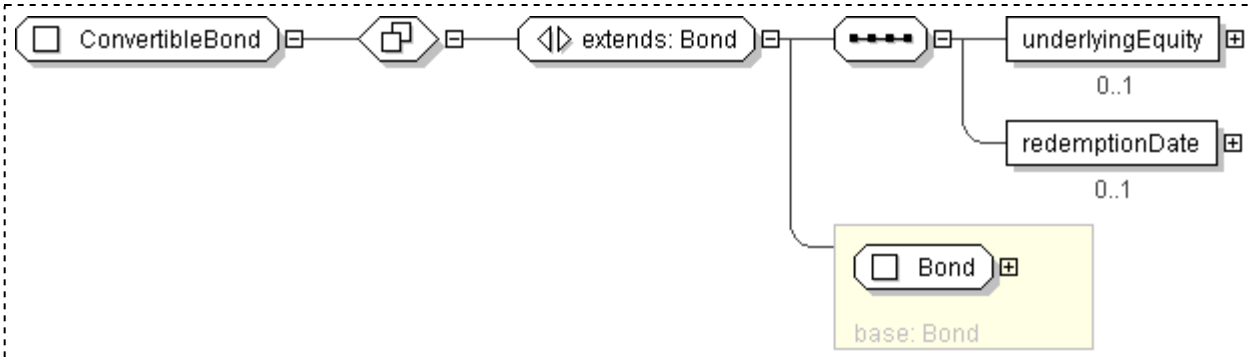
Name	ConvertibleBond
Used by (from the same schema document)	Element convertibleBond
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange
    for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'

   Start Choice [0..1]
 'Specifies the issuer name of a fixed income security or convertible bond. This
 name can either be explicitly stated, or specified as an href into another element
 of the document, such as the obligor'
```

```
      <issuerName> xsd:string </issuerName> [1]
      <issuerPartyReference> PartyReference </issuerPartyReference> [1]
   End Choice
   <seniority> CreditSeniority </seniority> [0..1]
   'The repayment precedence of a debt instrument.'

   <couponType> CouponType </couponType> [0..1]
   'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-
   coupon.'

   <couponRate> xsd:decimal </couponRate> [0..1]
   'Specifies the coupon rate (expressed in percentage) of a fixed income security
   or convertible bond.'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of a security becomes due and payable.'

   <parValue> xsd:decimal </parValue> [0..1]
   'Specifies the nominal amount of a fixed income security or convertible bond.'

   <faceAmount> xsd:decimal </faceAmount> [0..1]
   'Specifies the total amount of the issue. Corresponds to the par value
   multiplied by the number of issued security.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
   'Specifies the frequency at which the bond pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
   'The day count basis for the bond.'

   <underlyingEquity> EquityAsset </underlyingEquity> [0..1]
   'Specifies the equity in which the convertible bond can be converted.'

   <redemptionDate> xsd:date </redemptionDate> [0..1]
   'Earlier date between the convertible bond put dates and its maturity date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ConvertibleBond">
   <xsd:complexContent>
      <xsd:extension base=" Bond ">
         <xsd:sequence>
            <xsd:element name="underlyingEquity" type=" EquityAsset " minOccurs="0"/>
            <xsd:element name="redemptionDate" type=" xsd:date " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CouponType

[Table of contents]

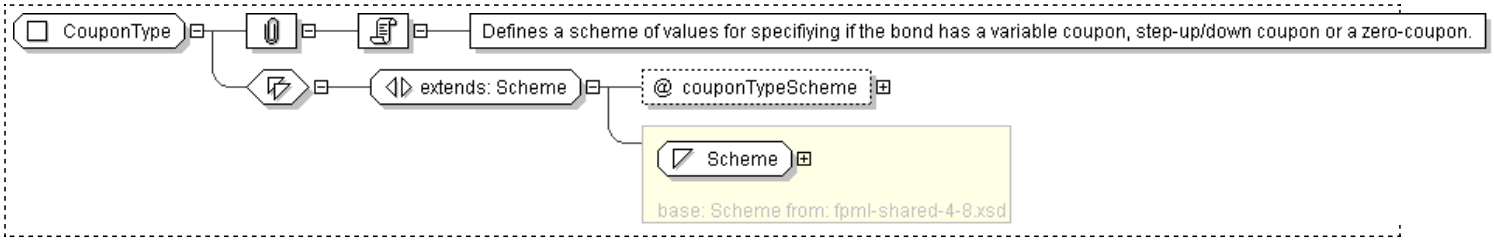
Super-types:	Scheme < CouponType (by extension)
Sub-types:	None

Name	CouponType
Used by (from the same schema document)	Model Group BondContent.model
Abstract	no
Documentation	Defines a scheme of values for specifiying if the bond has a variable coupon, step-up/down coupon or a zero-coupon.

XML Instance Representation

```
<...
 couponTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CouponType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="couponTypeScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/coupon-type"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Deposit

[Table of contents]

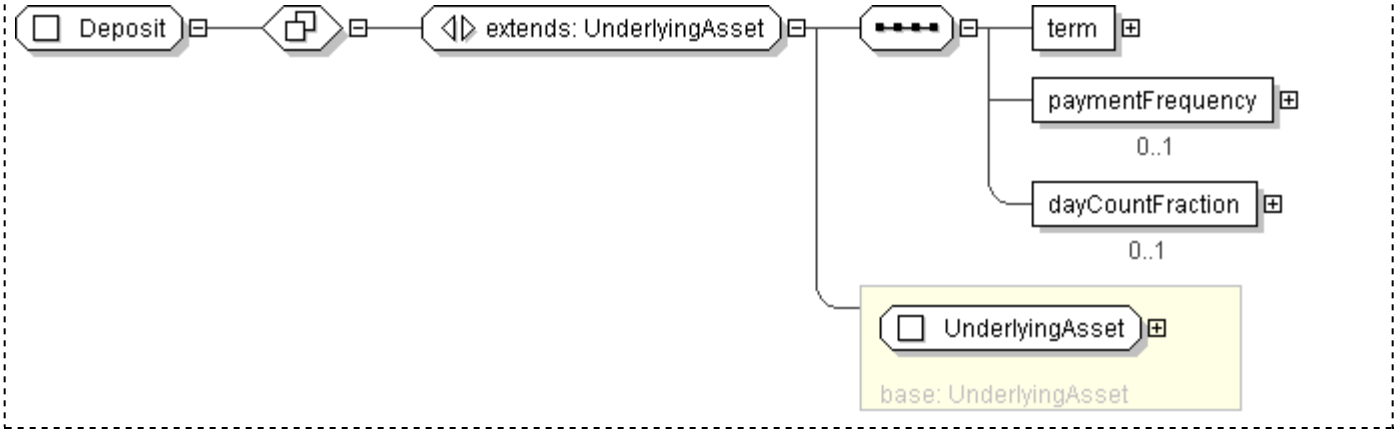
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < Deposit (by extension)
Sub-types:	None

Name	Deposit
Used by (from the same schema document)	Element deposit
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <term> Period </term> [1]
    'Specifies the term of the deposit, e.g. 5Y.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the deposit pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the deposit.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Deposit">
   <xsd:complexContent>
      <xsd:extension base=" UnderlyingAsset ">
         <xsd:sequence>
            <xsd:element name="term" type=" Period "/>
            <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendPayout

[Table of contents]

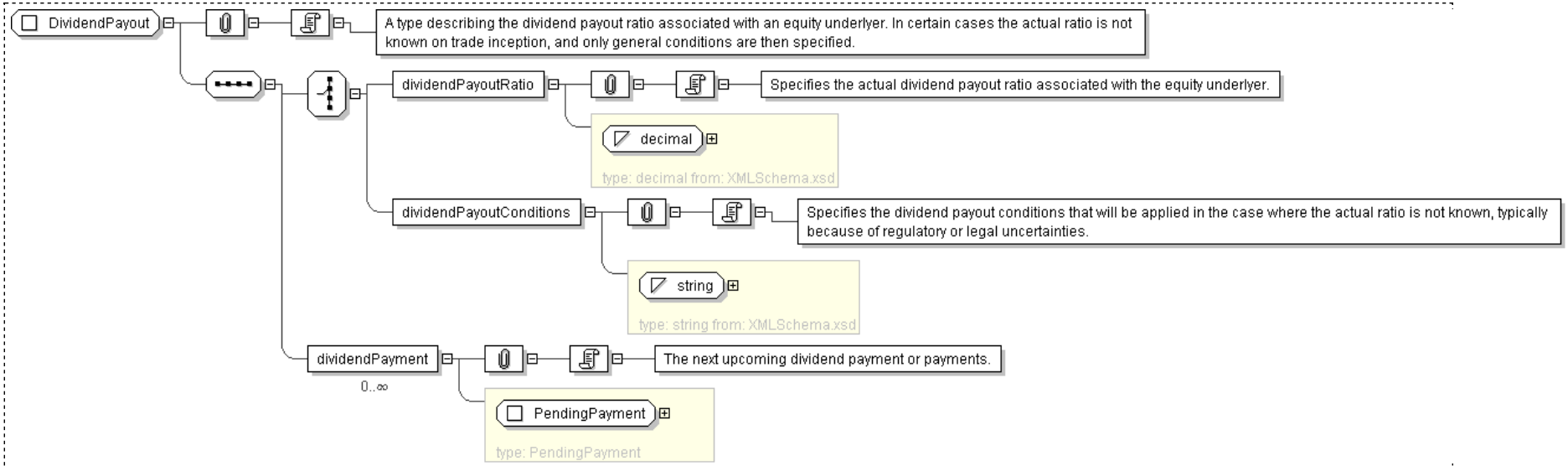
Super-types:	None
Sub-types:	None

Name	DividendPayout
Used by (from the same schema document)	Complex Type BasketConstituent , Complex Type SingleUnderlyer
Abstract	no
Documentation	A type describing the dividend payout ratio associated with an equity underlyer. In certain cases the actual ratio is not known on trade inception, and only general conditions are then specified.

XML Instance Representation

```
<...>
  Start Choice [1]
    <dividendPayoutRatio> xsd:decimal </dividendPayoutRatio> [1]
    'Specifies the actual dividend payout ratio associated with the equity underlyer.'

    <dividendPayoutConditions> xsd:string </dividendPayoutConditions> [1]
    'Specifies the dividend payout conditions that will be applied in the case where the actual ratio is not known, typically because of regulatory
    or legal uncertainties.'

  End Choice
  <dividendPayment> PendingPayment </dividendPayment> [0..*]
  'The next upcoming dividend payment or payments.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPayout">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="dividendPayoutRatio" type=" xsd:decimal "/>
      <xsd:element name="dividendPayoutConditions" type=" xsd:string "/>
```

```
      </xsd:choice>
      <xsd:element name="dividendPayment" type=" PendingPayment " minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityAsset

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < EquityAsset (by extension)
Sub-types:	None

Name	EquityAsset
Used by (from the same schema document)	Complex Type ConvertibleBond , Element equity
Abstract	no
Documentation	An exchange traded equity asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange
    for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'

</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="EquityAsset">
   <xsd:complexContent>
      <xsd:extension base=" ExchangeTraded "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeTraded

[Table of contents]

Super-types: Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension)

Sub-types:

- Bond (by extension)
  - ConvertibleBond (by extension)
- EquityAsset (by extension)
- ExchangeTradedCalculatedPrice (by extension)
  - ExchangeTradedFund (by extension)
  - Index (by extension)
- ExchangeTradedContract (by extension)
- Future (by extension)

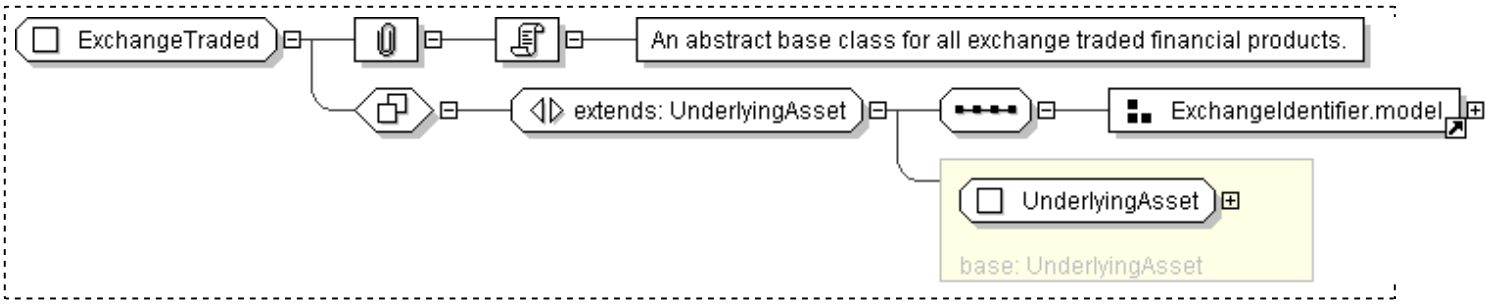
Name	ExchangeTraded
Abstract	yes
Documentation	An abstract base class for all exchange traded financial products.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in
    greater detail. In case of inconsistency between the terms of the simple product
    and those of the detailed definition, the values in the simple product override
    those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange for
    the future is different than the one for the option. The options Exchange is
    referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeTraded" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" UnderlyingAsset ">
      <xsd:sequence>
        <xsd:group ref=" ExchangeIdentifier.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeTradedCalculatedPrice

[Table of contents]

Super-types: Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < ExchangeTradedCalculatedPrice (by extension)

Sub-types:

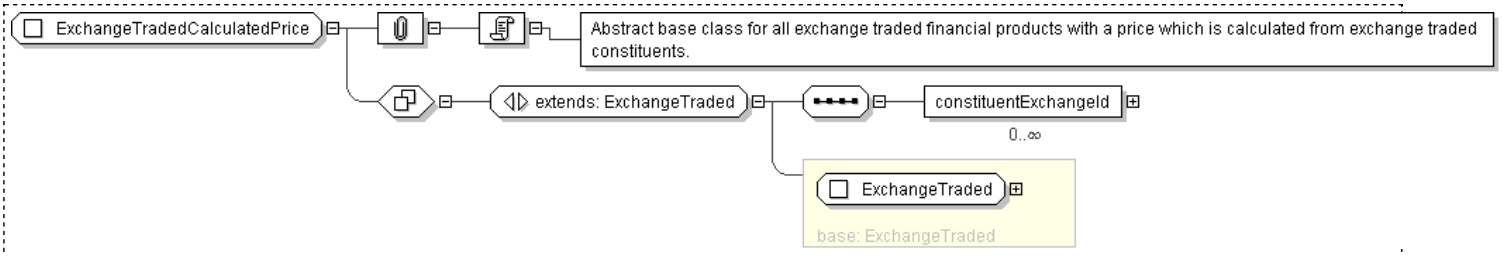
- ExchangeTradedFund (by extension)
- Index (by extension)

Name	ExchangeTradedCalculatedPrice
Abstract	yes
Documentation	Abstract base class for all exchange traded financial products with a price which is calculated from exchange traded constituents.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural
    payoff. The term \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail. In case of
    inconsistency between the terms of the simple product and those of the detailed definition, the values in the simple
    product override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not present then the exchange shall be the
    primary exchange on which listed futures and options on the underlying are listed. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option contract is listed. This is to address
    the case where the reference exchange for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options Exchange Adjustment.'

   <constituentExchangeId> ExchangeId </constituentExchangeId> [0..*]
    'Identification of all the exchanges where constituents are traded. The term \"Exchange\" is assumed to have the
    meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeTradedCalculatedPrice" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" ExchangeTraded ">
      <xsd:sequence>
        <xsd:element name="constituentExchangeId" type=" ExchangeId " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeTradedContract

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < ExchangeTradedContract (by extension)
Sub-types:	None

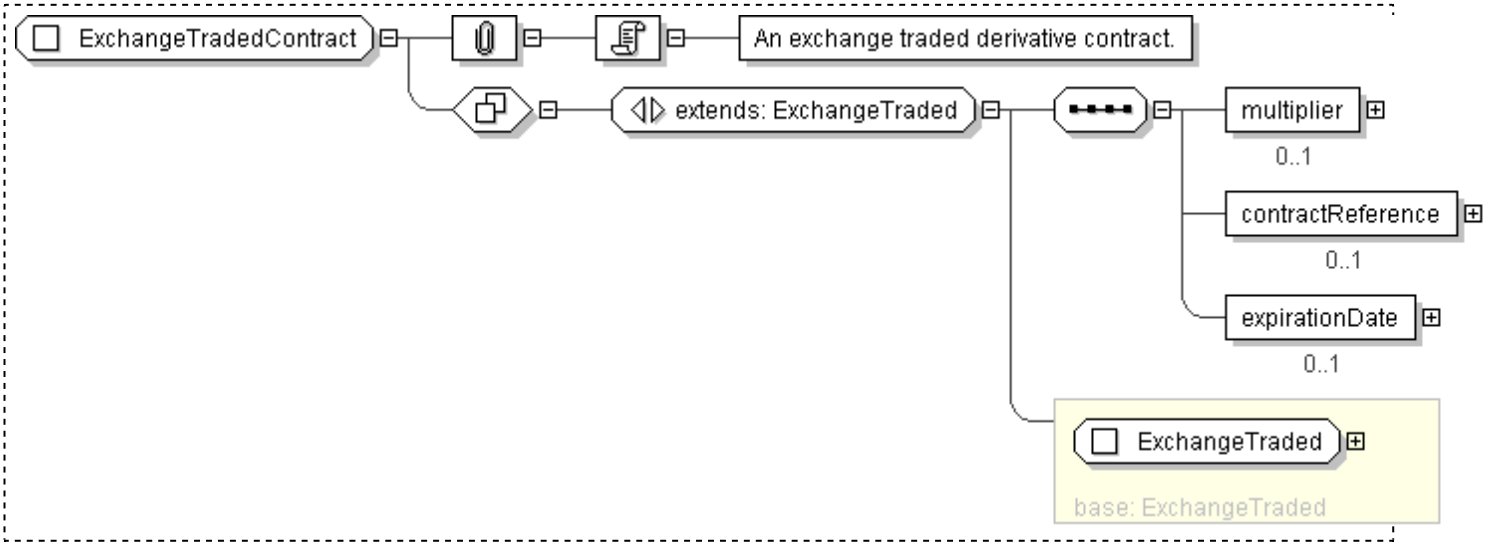
Name	ExchangeTradedContract
Abstract	no
Documentation	An exchange traded derivative contract.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private
   identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the
   purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
   to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction
   exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in
   greater detail. In case of inconsistency between the terms of the simple product
   and those of the detailed definition, the values in the simple product override
   those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
   'A short form unique identifier for a related exchange. If the element is not
   present then the exchange shall be the primary exchange on which listed futures
   and options on the underlying are listed. The term \"Exchange\" is assumed to
   have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
   'A short form unique identifier for an exchange on which the reference option
   contract is listed. This is to address the case where the reference exchange
   for the future is different than the one for the option. The options Exchange
   is referenced on share options when Merger Elections are selected as Options
   Exchange Adjustment.'

   <multiplier> xsd:positiveInteger </multiplier> [0..1]
   'Specifies the contract multiplier that can be associated with the number of
   units.'

   <contractReference> xsd:string </contractReference> [0..1]
   'Specifies the contract that can be referenced, besides the undelyer type.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [0..1]
```

```
    'The date when the contract expires.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeTradedContract">
   <xsd:complexContent>
      <xsd:extension base=" ExchangeTraded ">
         <xsd:sequence>
            <xsd:element name="multiplier" type=" xsd:positiveInteger " minOccurs="0"/>
            <xsd:element name="contractReference" type=" xsd:string " minOccurs="0"/>
            <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeTradedFund

[Table of contents]

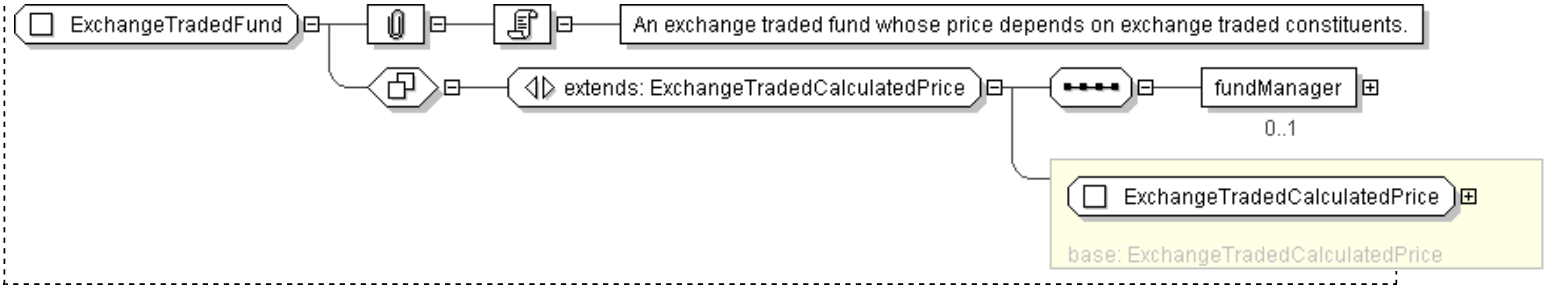
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < ExchangeTradedCalculatedPrice (by extension) < ExchangeTradedFund (by extension)
Sub-types:	None

Name	ExchangeTradedFund
Used by (from the same schema document)	Element exchangeTradedFund
Abstract	no
Documentation	An exchange traded fund whose price depends on exchange traded constituents.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of
   calculating a contractural payoff. The term \"Exchange\" is assumed to have the meaning as
   defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater
   detail. In case of inconsistency between the terms of the simple product and those of the
   detailed definition, the values in the simple product override those in the detailed
   definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
   'A short form unique identifier for a related exchange. If the element is not present then
   the exchange shall be the primary exchange on which listed futures and options on the
   underlying are listed. The term \"Exchange\" is assumed to have the meaning as defined in
   the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
   'A short form unique identifier for an exchange on which the reference option contract is
   listed. This is to address the case where the reference exchange for the future is different
   than the one for the option. The options Exchange is referenced on share options when Merger
   Elections are selected as Options Exchange Adjustment.'

   <constituentExchangeId> ExchangeId </constituentExchangeId> [0..*]
   'Identification of all the exchanges where constituents are traded. The term \"Exchange\" is
   assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <fundManager> xsd:string </fundManager> [0..1]
   'Specifies the fund manager that is in charge of the fund.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeTradedFund">
  <xsd:complexContent>
    <xsd:extension base=" ExchangeTradedCalculatedPrice ">
      <xsd:sequence>
        <xsd:element name="fundManager" type=" xsd:string " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FacilityType

[Table of contents]

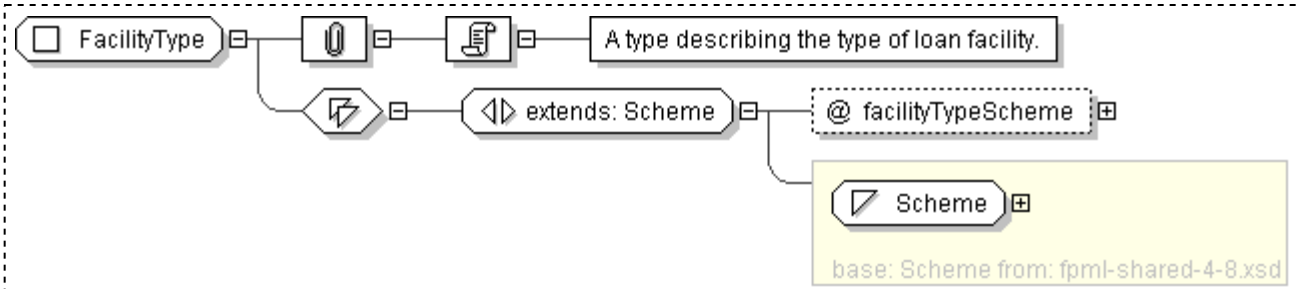
Super-types:	Scheme < FacilityType (by extension)
Sub-types:	None

Name	FacilityType
Used by (from the same schema document)	Complex Type Loan
Abstract	no
Documentation	A type describing the type of loan facility.

XML Instance Representation

```
<...
 facilityTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FacilityType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="facilityTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/facility-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Future

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < Future (by extension)
Sub-types:	None

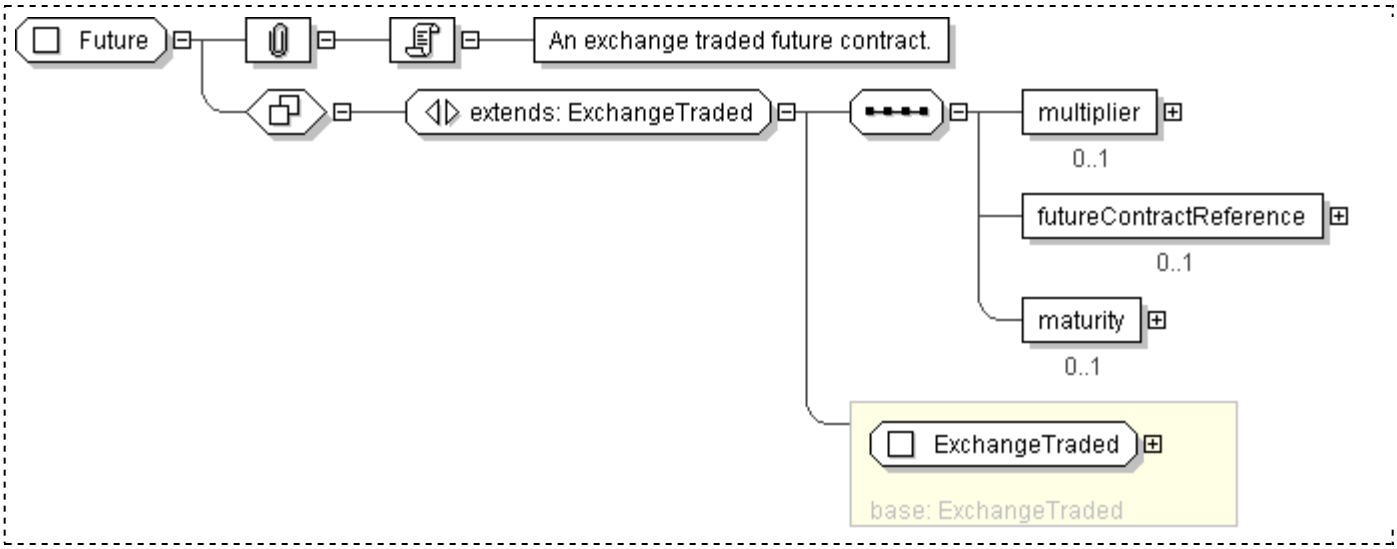
Name	Future
Used by (from the same schema document)	Element future
Abstract	no
Documentation	An exchange traded future contract.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange
    for the future is different than the one for the option. The options Exchange
    is referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'

   <multiplier> xsd:positiveInteger </multiplier> [0..1]
    'Specifies the contract multiplier that can be associated with the number of
    units.'
```

```
<futureContractReference> xsd:string </futureContractReference> [0..1]
'Specifies the future contract that can be referenced, besides the equity or index reference defined as part of the UnderlyerAsset type.'

<maturity> xsd:date </maturity> [0..1]
'The date when the future contract expires.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Future">
  <xsd:complexContent>
    <xsd:extension base=" ExchangeTraded ">
      <xsd:sequence>
        <xsd:element name="multiplier" type=" xsd:positiveInteger " minOccurs="0"/>
        <xsd:element name="futureContractReference" type=" xsd:string "
        minOccurs="0"/>
        <xsd:element name="maturity" type=" xsd:date " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FutureId

[Table of contents]

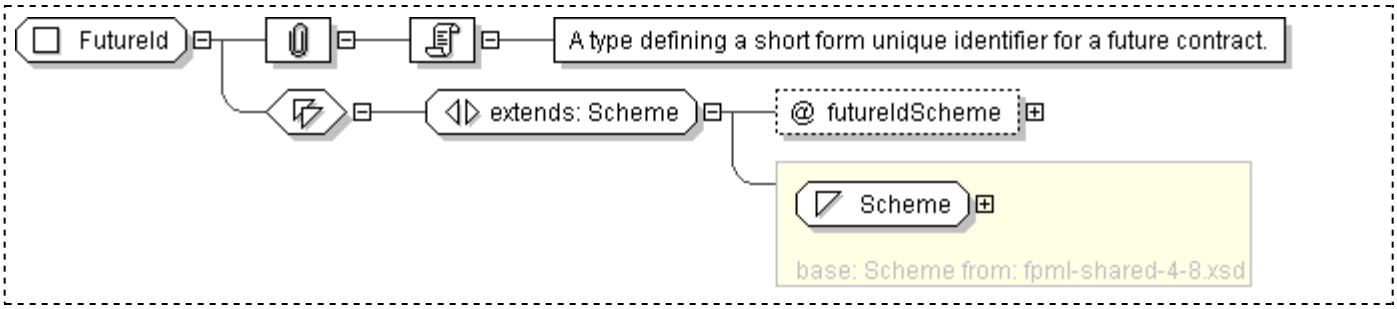
Super-types:	Scheme < FutureId (by extension)
Sub-types:	None

Name	FutureId
Used by (from the same schema document)	Complex Type Index
Abstract	no
Documentation	A type defining a short form unique identifier for a future contract.

XML Instance Representation

```
<...
 futureIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FutureId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="futureIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxConversion

[Table of contents]

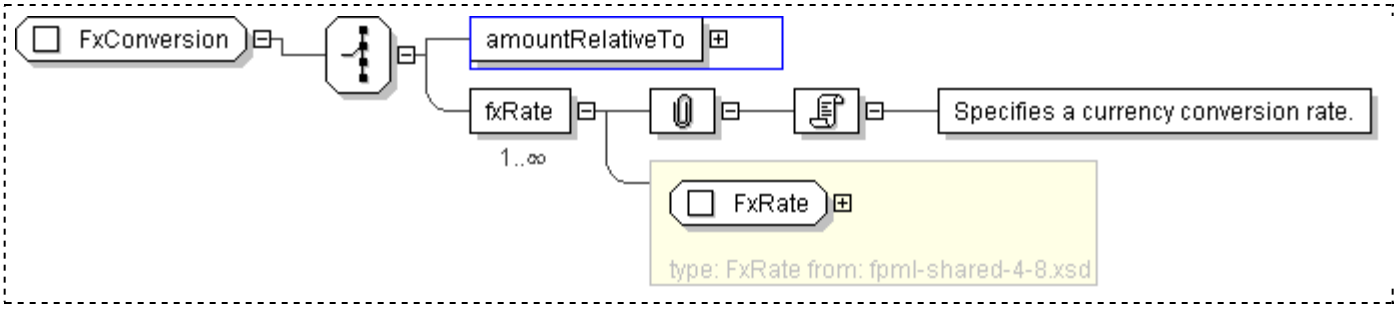
Super-types:	None
Sub-types:	None

Name	FxConversion
Used by (from the same schema document)	Model Group EquityPrice.model
Abstract	no

XML Instance Representation

```
<...>
   Start Choice [1]
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      <fxRate> FxRate </fxRate> [1..*]
      'Specifies a currency conversion rate.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxConversion">
   <xsd:choice>
      <xsd:element name="amountRelativeTo" type=" AmountReference "/>
      <xsd:element name="fxRate" type=" FxRate " maxOccurs="unbounded"/>
   </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxRateAsset

[Table of contents]

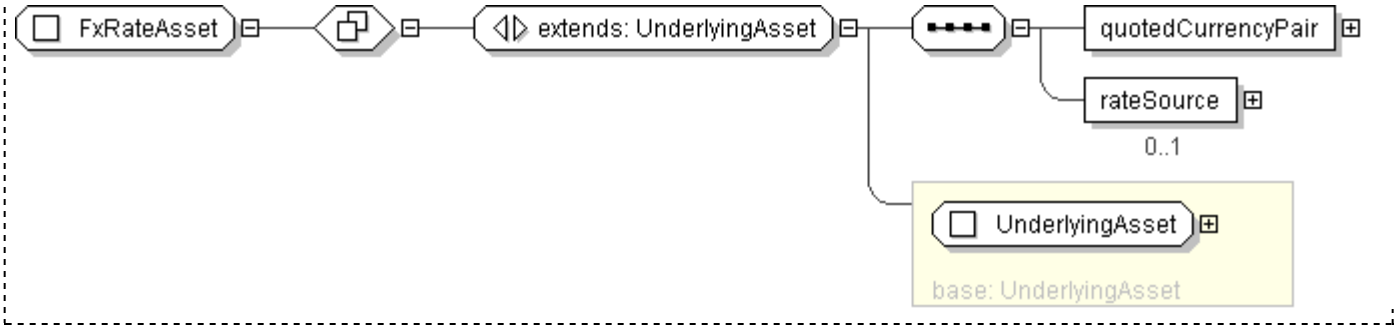
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < FxRateAsset (by extension)
Sub-types:	None

Name	FxRateAsset
Used by (from the same schema document)	Element fxRate
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
    'Defines the two currencies for an FX trade and the quotation relationship
    between the two currencies.'

   <rateSource> FxSpotRateSource </rateSource> [0..1]
    'Defines the source of the FX rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxRateAsset">
  <xsd:complexContent>
    <xsd:extension base=" UnderlyingAsset ">
      <xsd:sequence>
        <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
        <xsd:element name="rateSource" type=" FxSpotRateSource " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IdentifiedAsset

[Table of contents]

Super-types: Asset < IdentifiedAsset (by extension)

Sub-types:

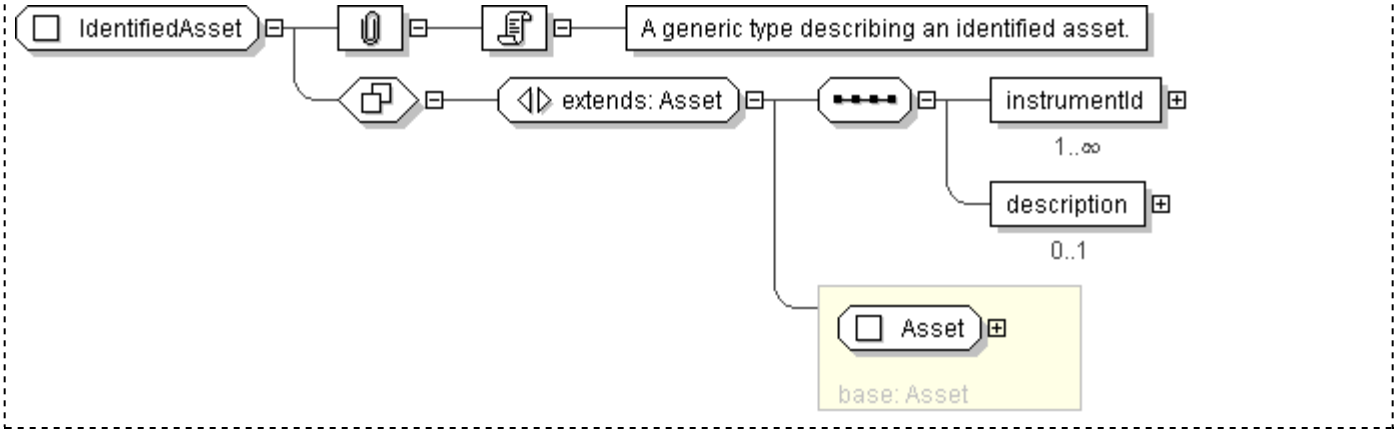
- Cash (by extension)
- Commodity (by extension)
- UnderlyingAsset (by extension)
  - Deposit (by extension)
  - ExchangeTraded (by extension)
    - Bond (by extension)
      - ConvertibleBond (by extension)
    - EquityAsset (by extension)
    - ExchangeTradedCalculatedPrice (by extension)
      - ExchangeTradedFund (by extension)
      - Index (by extension)
    - ExchangeTradedContract (by extension)
    - Future (by extension)
  - FxRateAsset (by extension)
  - Loan (by extension)
  - Mortgage (by extension)
  - MutualFund (by extension)
  - RateIndex (by extension)
  - SimpleCreditDefaultSwap (by extension)
  - SimpleFra (by extension)
  - SimpleIRSwap (by extension)

Name	IdentifiedAsset
Abstract	yes
Documentation	A generic type describing an identified asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IdentifiedAsset" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Asset ">
         <xsd:sequence>
            <xsd:element name="instrumentId" type=" InstrumentId "
            maxOccurs="unbounded"/>
            <xsd:element name="description" type=" xsd:string " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Index

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < ExchangeTraded (by extension) < ExchangeTradedCalculatedPrice (by extension) < Index (by extension)
Sub-types:	None

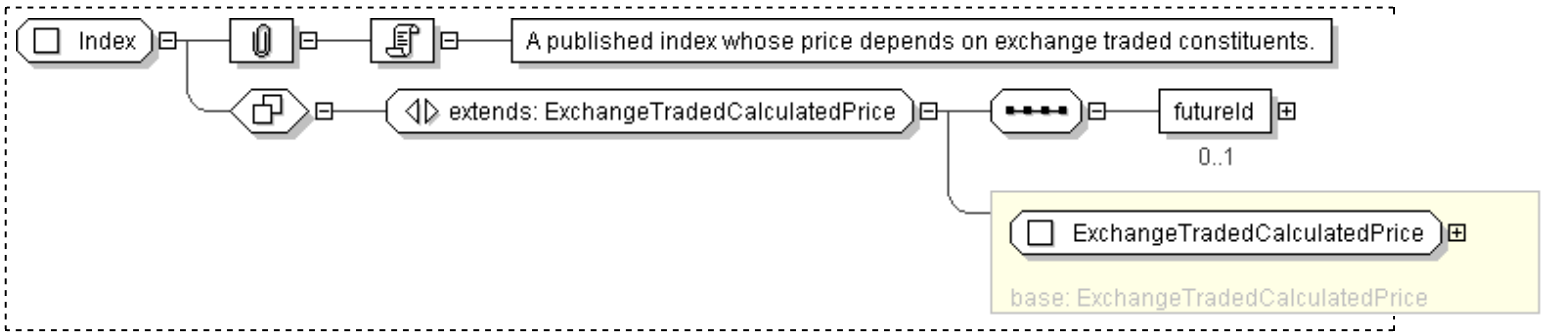
Name	Index
Used by (from the same schema document)	Element index
Abstract	no
Documentation	A published index whose price depends on exchange traded constituents.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in
    greater detail. In case of inconsistency between the terms of the simple product
    and those of the detailed definition, the values in the simple product override
    those in the detailed definition.'

   <relatedExchangeId> ExchangeId </relatedExchangeId> [0..*]
    'A short form unique identifier for a related exchange. If the element is not
    present then the exchange shall be the primary exchange on which listed futures
    and options on the underlying are listed. The term \"Exchange\" is assumed to
    have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <optionsExchangeId> ExchangeId </optionsExchangeId> [0..*]
    'A short form unique identifier for an exchange on which the reference option
    contract is listed. This is to address the case where the reference exchange for
    the future is different than the one for the option. The options Exchange is
    referenced on share options when Merger Elections are selected as Options
    Exchange Adjustment.'

   <constituentExchangeId> ExchangeId </constituentExchangeId> [0..*]
    'Identification of all the exchanges where constituents are traded. The term
    \"Exchange\" is assumed to have the meaning as defined in the ISDA 2002 Equity
    Derivatives Definitions.'

   <futureId> FutureId </futureId> [0..1]
    'A short form unique identifier for the reference future contract in the case of
    an index underlyer.'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Index">
   <xsd:complexContent>
      <xsd:extension base=" ExchangeTradedCalculatedPrice ">
         <xsd:sequence>
            <xsd:element name="futureId" type=" FutureId " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Lien

[Table of contents]

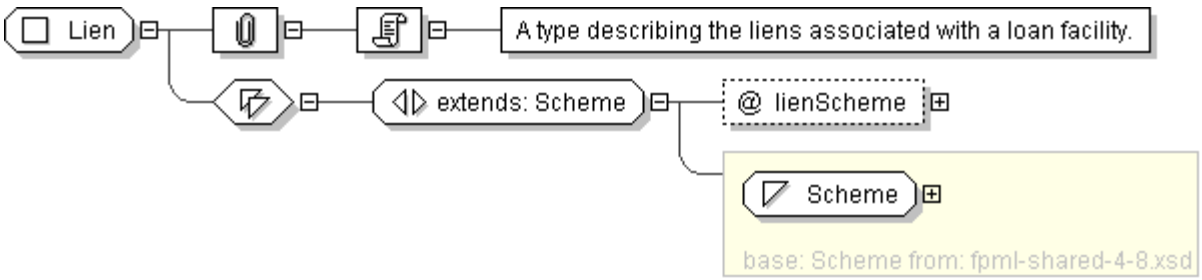
Super-types:	Scheme < Lien (by extension)
Sub-types:	None

Name	Lien
Used by (from the same schema document)	Complex Type Loan
Abstract	no
Documentation	A type describing the liens associated with a loan facility.

XML Instance Representation

```
<...
 lienScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Lien">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="lienScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/designated-priority"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Loan

[Table of contents]

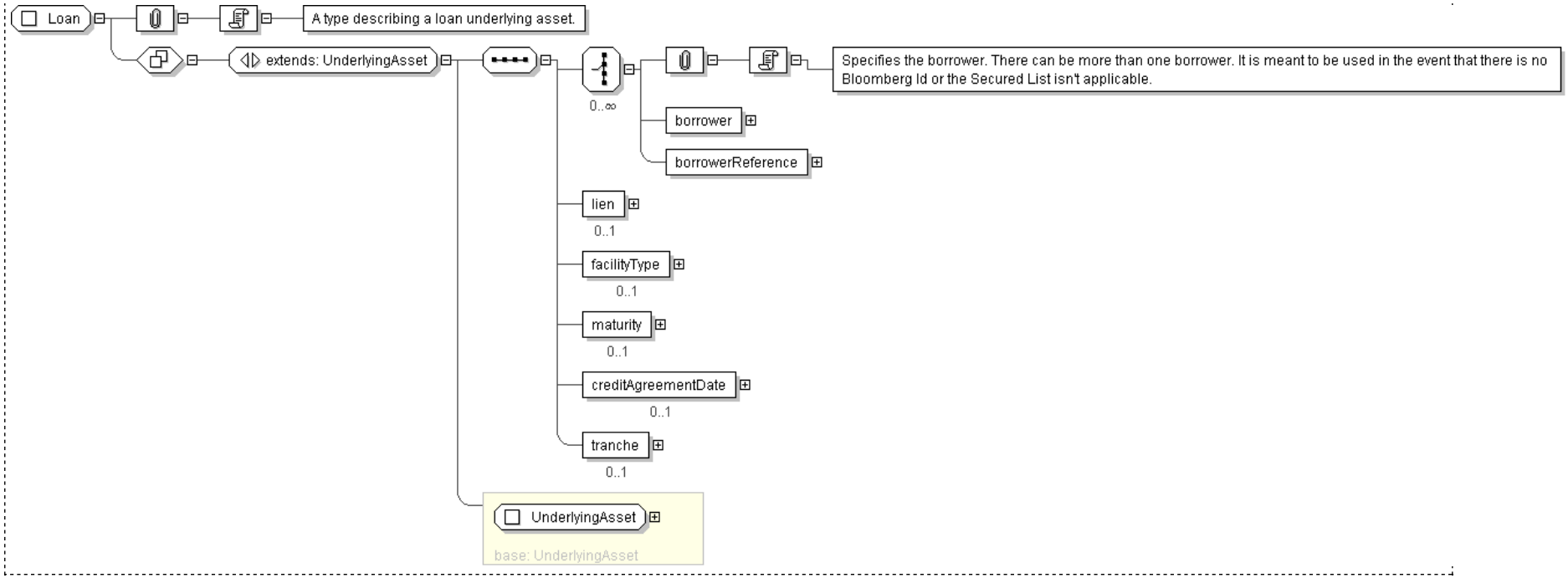
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < Loan (by extension)
Sub-types:	None

Name	Loan
Used by (from the same schema document)	Element loan
Abstract	no
Documentation	A type describing a loan underlying asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural payoff. The term \"Exchange\" is
   assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product in greater detail. In case of inconsistency between the terms of the
   simple product and those of the detailed definition, the values in the simple product override those in the detailed definition.'

   Start Choice [0..*]
'Specifies the borrower. There can be more than one borrower. It is meant to be used in the event that there is no Bloomberg Id or the Secured List
isn\'t applicable.'


      <borrower> LegalEntity </borrower> [1]
      <borrowerReference> LegalEntityReference </borrowerReference> [1]
   End Choice
   <lien> Lien </lien> [0..1]
   'Specifies the seniority level of the lien.'

   <facilityType> FacilityType </facilityType> [0..1]
   'The type of loan facility (letter of credit, revolving, ...).'

   <maturity> xsd:date </maturity> [0..1]
   'The date when the principal amount of the loan becomes due and payable.'

   <creditAgreementDate> xsd:date </creditAgreementDate> [0..1]
   'The credit agreement date is the closing date (the date where the agreement has been signed) for the loans in the credit agreement. Funding of the
   facilities occurs on (or sometimes a little after) the Credit Agreement date. This underlyer attribute is used to help identify which of the
   company\'s outstanding loans are being referenced by knowing to which credit agreement it belongs. ISDA Standards Terms Supplement term: Date of
   Original Credit Agreement.'

   <tranche> UnderlyingAssetTranche </tranche> [0..1]
   'The loan tranche that is subject to the derivative transaction. It will typically be referenced as the Bloomberg tranche number. ISDA Standards
   Terms Supplement term: Bloomberg Tranche Number.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Loan">
  <xsd:complexContent>
    <xsd:extension base=" UnderlyingAsset ">
      <xsd:sequence>
        <xsd:choice minOccurs="0" maxOccurs="unbounded">
          <xsd:element name="borrower" type=" LegalEntity "/>
          <xsd:element name="borrowerReference" type=" LegalEntityReference "/>
        </xsd:choice>
        <xsd:element name="lien" type=" Lien " minOccurs="0"/>
        <xsd:element name="facilityType" type=" FacilityType " minOccurs="0"/>
        <xsd:element name="maturity" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="creditAgreementDate" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="tranche" type=" UnderlyingAssetTranche " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Mortgage

[Table of contents]

Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < Mortgage (by extension)
Sub-types:	None

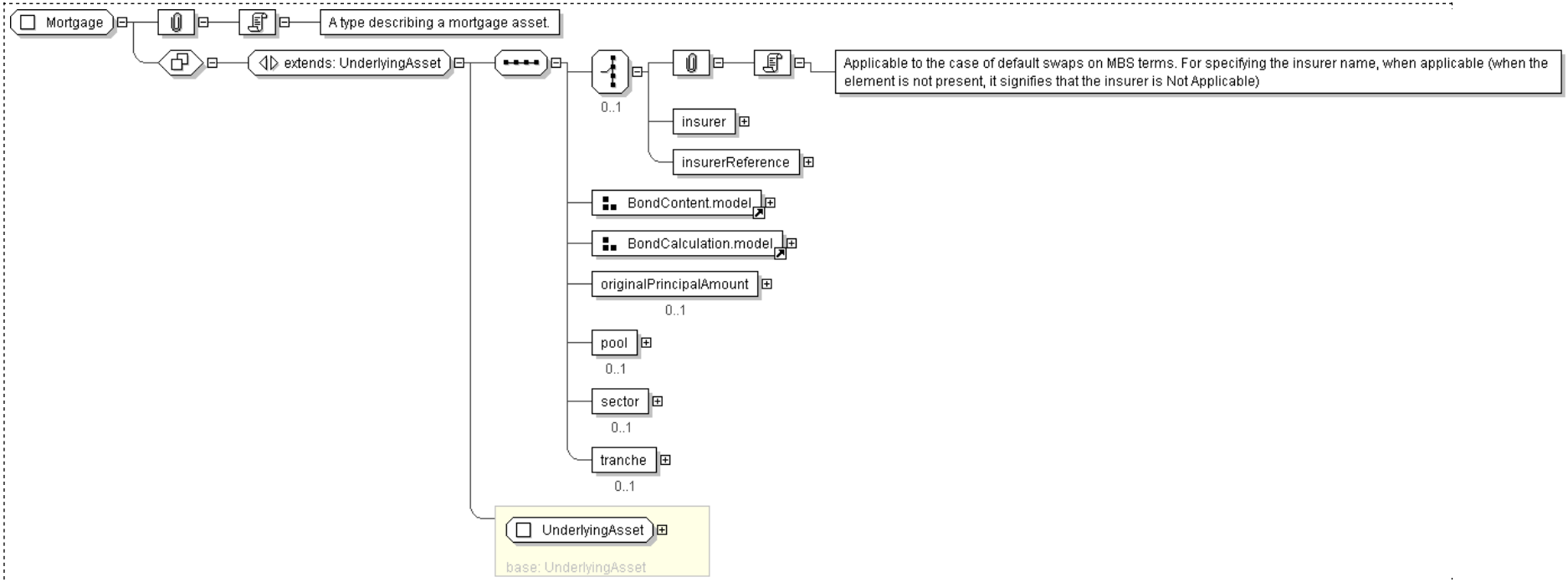
Name	Mortgage
Used by (from the same schema document)	Element mortgage
Abstract	no
Documentation	A type describing a mortgage asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product override those in the detailed definition.'

   Start Choice [0..1]
 'Applicable to the case of default swaps on MBS terms. For specifying the insurer name, when applicable (when the element is not present, it signifies that
 the insurer is Not Applicable)'


      <insurer> LegalEntity </insurer> [1]
      <insurerReference> LegalEntityReference </insurerReference> [1]
   End Choice
   Start Choice [0..1]
 'Specifies the issuer name of a fixed income security or convertible bond. This name can either be explicitly stated, or specified as an href into another
 element of the document, such as the obligor'


      <issuerName> xsd:string </issuerName> [1]
      <issuerPartyReference> PartyReference </issuerPartyReference> [1]
   End Choice
   <seniority> CreditSeniority </seniority> [0..1]
    'The repayment precedence of a debt instrument.'

   <couponType> CouponType </couponType> [0..1]
    'Specifies if the bond has a variable coupon, step-up/down coupon or a zero-coupon.'

   <couponRate> xsd:decimal </couponRate> [0..1]
    'Specifies the coupon rate (expressed in percentage) of a fixed income security or convertible bond.'

   <maturity> xsd:date </maturity> [0..1]
    'The date when the principal amount of a security becomes due and payable.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the bond pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the bond.'

   <originalPrincipalAmount> xsd:decimal </originalPrincipalAmount> [0..1]
    'The initial issued amount of the mortgage obligation.'
```

```
<pool> AssetPool </pool> [0..1]
'The morgage pool that is underneath the mortgage obligation.'

<sector> MortgageSector </sector> [0..1]
'The sector classification of the mortgage obligation.'

<tranche> xsd:token </tranche> [0..1]
'The mortgage obligation tranche that is subject to the derivative transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Mortgage">
  <xsd:complexContent>
    <xsd:extension base=" UnderlyingAsset ">
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="insurer" type=" LegalEntity "/>
          <xsd:element name="insurerReference" type=" LegalEntityReference "/>
        </xsd:choice>
        <xsd:group ref=" BondContent.model "/>
        <xsd:group ref=" BondCalculation.model "/>
        <xsd:element name="originalPrincipalAmount" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="pool" type=" AssetPool " minOccurs="0"/>
        <xsd:element name="sector" type=" MortgageSector " minOccurs="0"/>
        <xsd:element name="tranche" type=" xsd:token " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MortgageSector

[Table of contents]

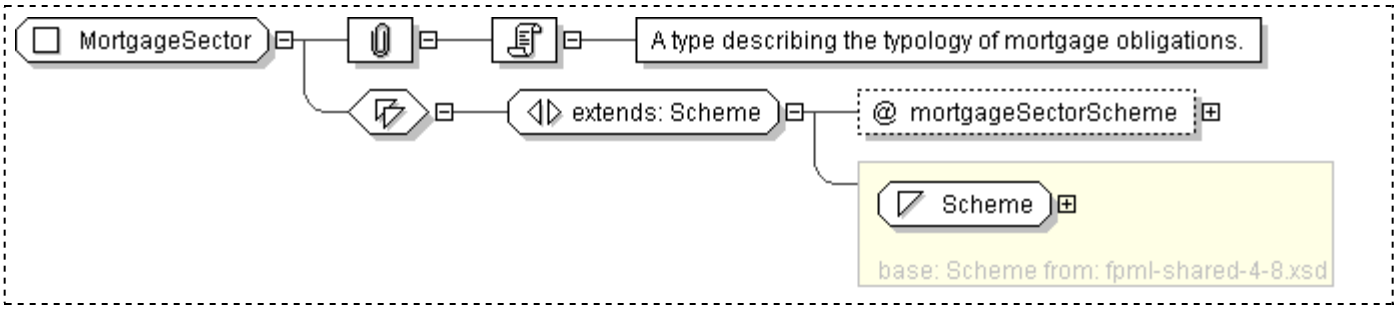
Super-types:	Scheme < MortgageSector (by extension)
Sub-types:	None

Name	MortgageSector
Used by (from the same schema document)	Complex Type Mortgage
Abstract	no
Documentation	A type describing the typology of mortgage obligations.

XML Instance Representation

```
<...
 mortgageSectorScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MortgageSector">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="mortgageSectorScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/mortgage-sector"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MutualFund

[Table of contents]

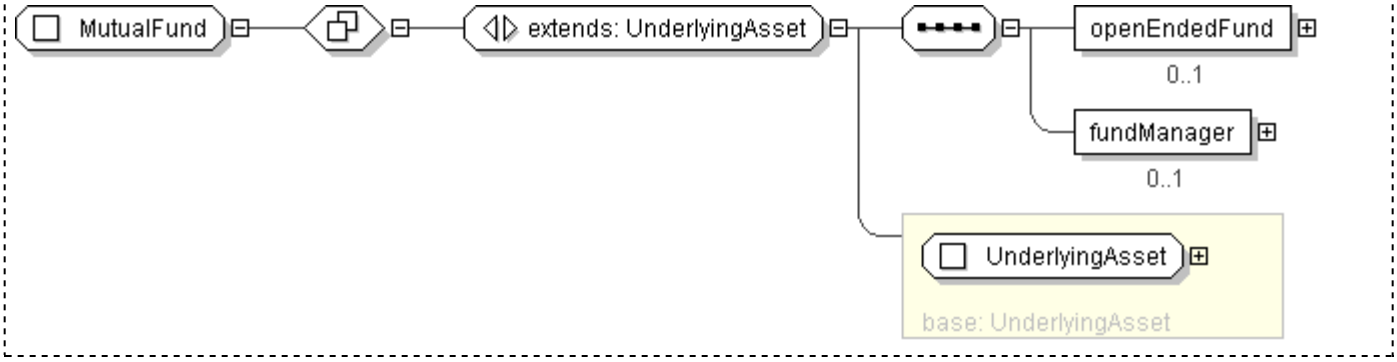
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < MutualFund (by extension)
Sub-types:	None

Name	MutualFund
Used by (from the same schema document)	Element mutualFund
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <openEndedFund> xsd:boolean </openEndedFund> [0..1]
    'Boolean indicator to specify whether the mutual fund is an open-ended mutual
    fund.'

   <fundManager> xsd:string </fundManager> [0..1]
    'Specifies the fund manager that is in charge of the fund.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MutualFund">
   <xsd:complexContent>
      <xsd:extension base=" UnderlyingAsset ">
         <xsd:sequence>
            <xsd:element name="openEndedFund" type=" xsd:boolean " minOccurs="0"/>
            <xsd:element name="fundManager" type=" xsd:string " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PendingPayment

[Table of contents]

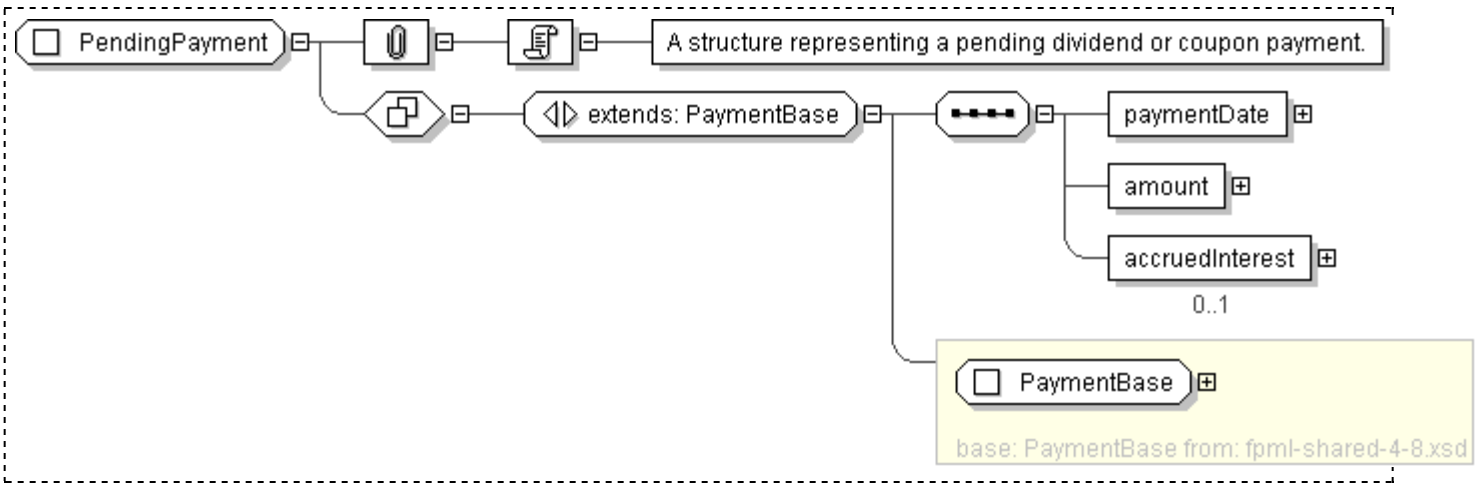
Super-types:	PaymentBase < PendingPayment (by extension)
Sub-types:	None

Name	PendingPayment
Used by (from the same schema document)	Complex Type BasketConstituent , Complex Type DividendPayout , Complex Type SingleUnderlyer
Abstract	no
Documentation	A structure representing a pending dividend or coupon payment.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <paymentDate> xsd:date </paymentDate> [1]
   'The date that the dividend or coupon is due.'

   <amount> Money </amount> [1]
   'The amount of the dividend or coupon payment. Value of dividends or coupon
   between ex and pay date. Stock: if we are between ex-date and pay-date and the
   dividend is payable under the swap, then this should be the ex-div amount * #
   of securities. Bond: regardless of where we are vis-a-vis resets: (coupon % *
   face of bonds on swap * (bond day count fraction using days last coupon pay
   date of the bond through today).'

   <accruedInterest> Money </accruedInterest> [0..1]
   'Accrued interest on the dividend or coupon payment. When the TRS is structured
   to pay a dividend or coupon on reset after payable date, you may earn interest
   on these amounts. This field indicates the interest accrued on dividend/coupon
   from pay date to statement date. This will only apply to a handful of
   agreements where dividendss are held to the next reset AND you receive/pay
   interest on unpaid amounts.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PendingPayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:element name="paymentDate" type=" xsd:date "/>
            <xsd:element name="amount" type=" Money "/>
            <xsd:element name="accruedInterest" type=" Money " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Price

[Table of contents]

Super-types:	None
Sub-types:	None

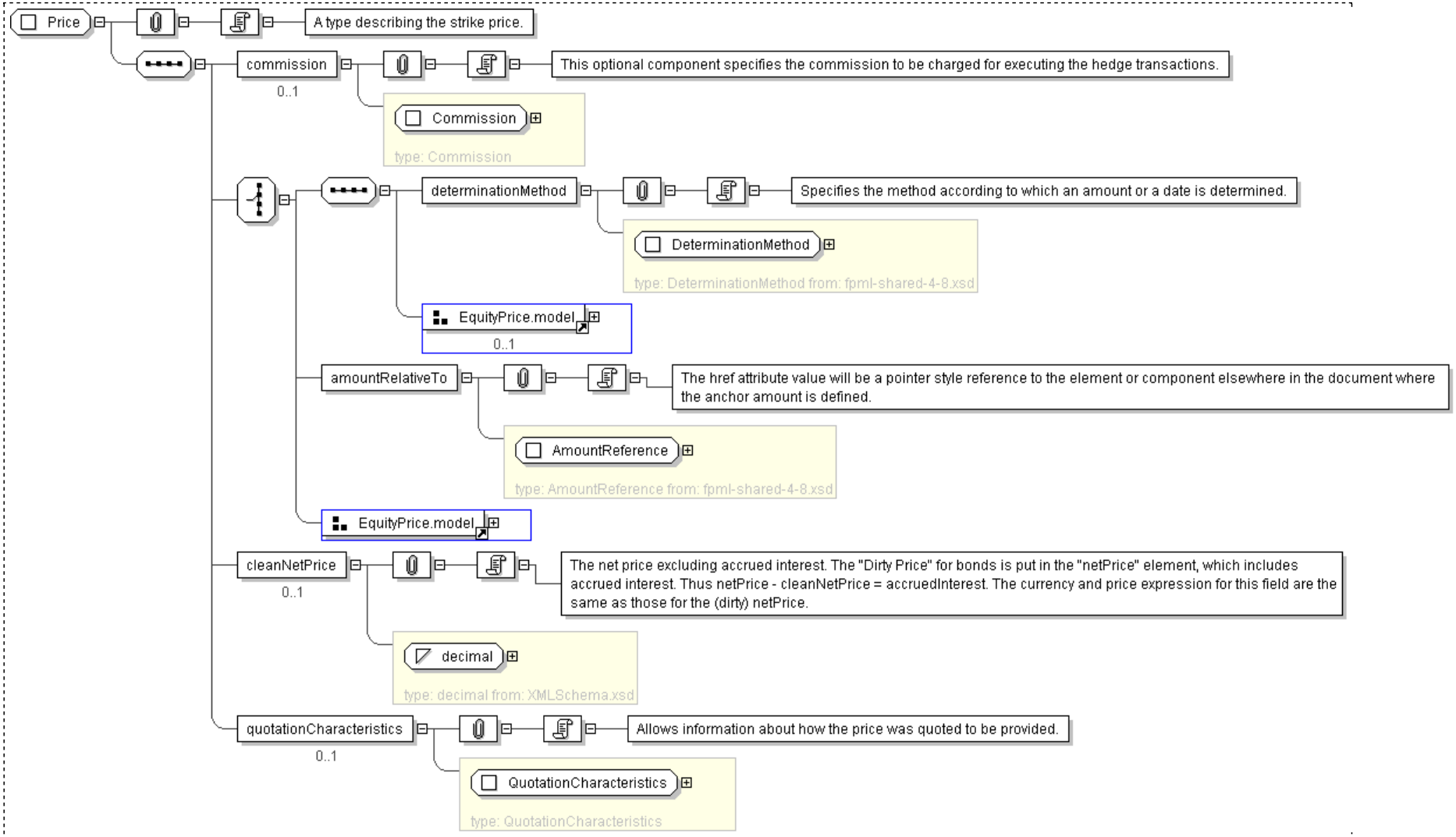
Name	Price
Used by (from the same schema document)	Complex Type BasketConstituent
Abstract	no
Documentation	A type describing the strike price.

XML Instance Representation

```
<...>
   <commission> Commission </commission> [0..1]
   'This optional component specifies the commission to be charged for executing the hedge transactions.'

   Start Choice [1]
      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      Start Group: EquityPrice.model [0..1]
         <grossPrice> ActualPrice </grossPrice> [0..1]
         'Specifies the price of the underlyer, before commissions.'

         <netPrice> ActualPrice </netPrice> [1]
         'Specifies the price of the underlyer, net of commissions.'

         <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
         'Specifies the accrued interest that are part of the dirty price in the case of a fixed income security or a convertible bond.
         Expressed in percentage of the notional.'

         <fxConversion> FxConversion </fxConversion> [0..1]
         'Specifies the currency conversion rate that applies to an amount. This rate can either be defined elsewhere in the document (case
         of a quanto swap), or explicitly described through this component.'

      End Group: EquityPrice.model
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'The href attribute value will be a pointer style reference to the element or component elsewhere in the document where the anchor
      amount is defined.'

      <grossPrice> ActualPrice </grossPrice> [0..1]
      'Specifies the price of the underlyer, before commissions.'

      <netPrice> ActualPrice </netPrice> [1]
      'Specifies the price of the underlyer, net of commissions.'

      <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
      'Specifies the accrued interest that are part of the dirty price in the case of a fixed income security or a convertible bond.
      Expressed in percentage of the notional.'

      <fxConversion> FxConversion </fxConversion> [0..1]
      'Specifies the currency conversion rate that applies to an amount. This rate can either be defined elsewhere in the document (case of
      a quanto swap), or explicitly described through this component.'

   End Choice
   <cleanNetPrice> xsd:decimal </cleanNetPrice> [0..1]
   'The net price excluding accrued interest. The \"Dirty Price\" for bonds is put in the \"netPrice\" element, which includes accrued
   interest. Thus netPrice - cleanNetPrice = accruedInterest. The currency and price expression for this field are the same as those for the
   (dirty) netPrice.'

   <quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..1]
   'Allows information about how the price was quoted to be provided.'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Price">
  <xsd:sequence>
    <xsd:element name="commission" type=" Commission " minOccurs="0"/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
        <xsd:group ref=" EquityPrice.model " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="amountRelativeTo" type=" AmountReference "/>
      <xsd:group ref=" EquityPrice.model "/>
    </xsd:choice>
    <xsd:element name="cleanNetPrice" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="quotationCharacteristics" type=" QuotationCharacteristics " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PriceQuoteUnits

[Table of contents]

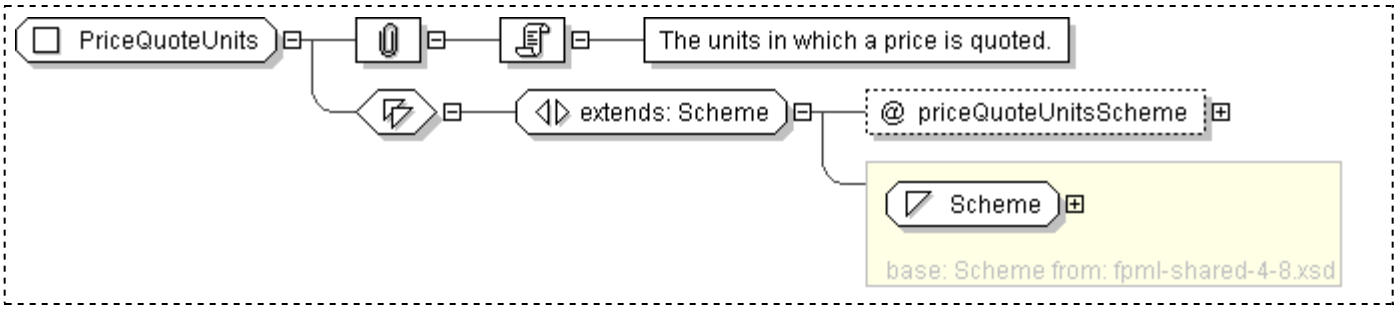
Super-types:	Scheme < PriceQuoteUnits (by extension)
Sub-types:	None

Name	PriceQuoteUnits
Used by (from the same schema document)	Model Group QuotationCharacteristics.model
Abstract	no
Documentation	The units in which a price is quoted.

XML Instance Representation

```
<...
 priceQuoteUnitsScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PriceQuoteUnits">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="priceQuoteUnitsScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/price-quote-units"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuantityUnit

[Table of contents]

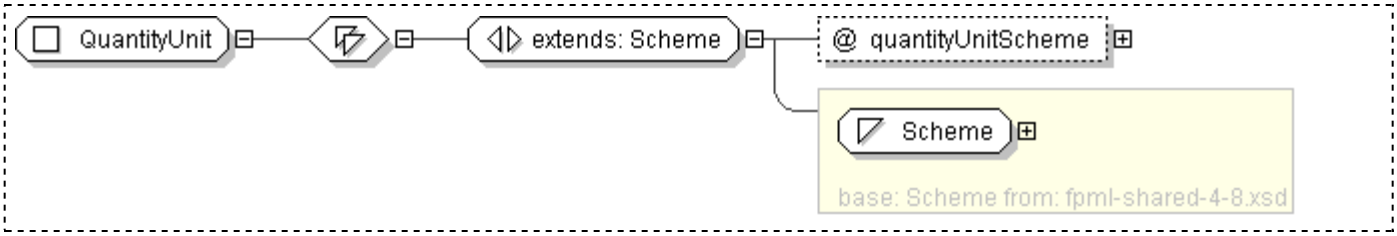
Super-types:	Scheme < QuantityUnit (by extension)
Sub-types:	None

Name	QuantityUnit
Used by (from the same schema document)	Model Group CommodityReferencePriceFramework.model
Abstract	no

XML Instance Representation

```
<...
 quantityUnitScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuantityUnit">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="quantityUnitScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/price-quote-units"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotationCharacteristics

[Table of contents]

Super-types:	None
Sub-types:	None

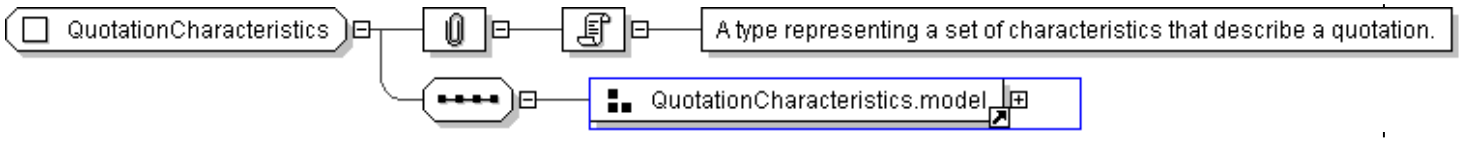
Name	QuotationCharacteristics
Used by (from the same schema document)	Complex Type Price
Abstract	no
Documentation	A type representing a set of characteristics that describe a quotation.

XML Instance Representation

```
<...>
   <measureType> AssetMeasureType </measureType> [0..1]
   'The type of the value that is measured. This could be an NPV, a cash flow, a
   clean price, etc.'

   <quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
   'The optional units that the measure is expressed in. If not supplied, this is
   assumed to be a price/value in currency units.'

   <side> QuotationSideEnum </side> [0..1]
   'The side (bid/mid/ask) of the measure.'

   <currency> Currency </currency> [0..1]
   'The optional currency that the measure is expressed in. If not supplied, this is
   defaulted from the reportingCurrency in the valuationScenarioDefinition.'

   <timing> QuoteTiming </timing> [0..1]
   'When during a day the quote is for. Typically, if this element is supplied, the
   QuoteLocation needs also to be supplied.'

   Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


      Start Choice [1]
         <businessCenter> BusinessCenter </businessCenter> [1]
         'A city or other business center.'

         <exchangeId> ExchangeId </exchangeId> [1]
         'The exchange (e.g. stock or futures exchange) from which the quote is
         obtained.'

      End Choice
   End Group: QuoteLocation.model
   <informationSource> InformationSource </informationSource> [0..*]
   'The information source where a published or displayed market rate will be
   obtained, e.g. Telerate Page 3750.'

   <time> xsd:dateTime </time> [0..1]
   'When the quote was observed or derived.'

   <valuationDate> xsd:date </valuationDate> [0..1]
   'When the quote was computed.'

   <expiryTime> xsd:dateTime </expiryTime> [0..1]
   'When does the quote cease to be valid.'

   <cashFlowType> CashflowType </cashFlowType> [0..1]
```

```
   'For cash flows, the type of the cash flows. Examples include: Coupon payment,
   Premium Fee, Settlement Fee, Brokerage Fee, etc.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotationCharacteristics">
   <xsd:sequence>
      <xsd:group ref=" QuotationCharacteristics.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuoteTiming

[Table of contents]

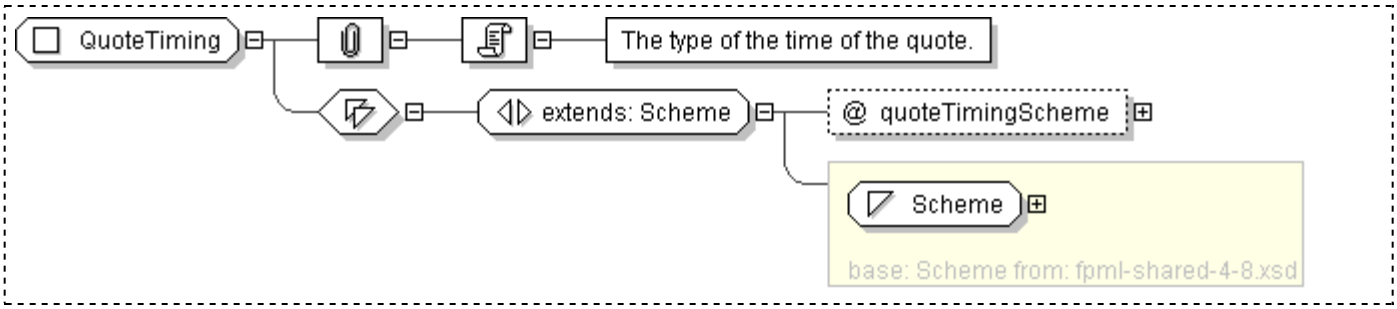
Super-types:	Scheme < QuoteTiming (by extension)
Sub-types:	None

Name	QuoteTiming
Used by (from the same schema document)	Model Group QuotationCharacteristics.model
Abstract	no
Documentation	The type of the time of the quote.

XML Instance Representation

```
<...
 quoteTimingScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuoteTiming">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="quoteTimingScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/quote-timing"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RateIndex

[Table of contents]

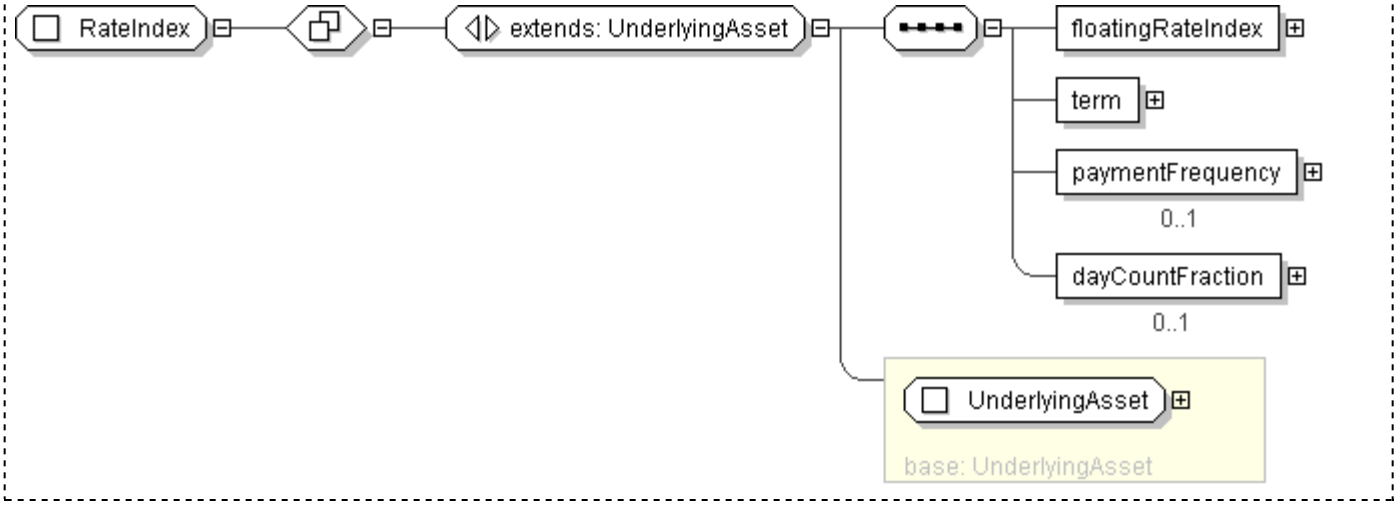
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < RateIndex (by extension)
Sub-types:	None

Name	RateIndex
Used by (from the same schema document)	Element rateIndex
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <term> Period </term> [1]
    'Specifies the term of the simple swap, e.g. 5Y.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the index pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the index.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RateIndex">
   <xsd:complexContent>
      <xsd:extension base=" UnderlyingAsset ">
         <xsd:sequence>
            <xsd:element name="floatingRateIndex" type=" FloatingRateIndex "/>
            <xsd:element name="term" type=" Period "/>
            <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SimpleCreditDefaultSwap

[Table of contents]

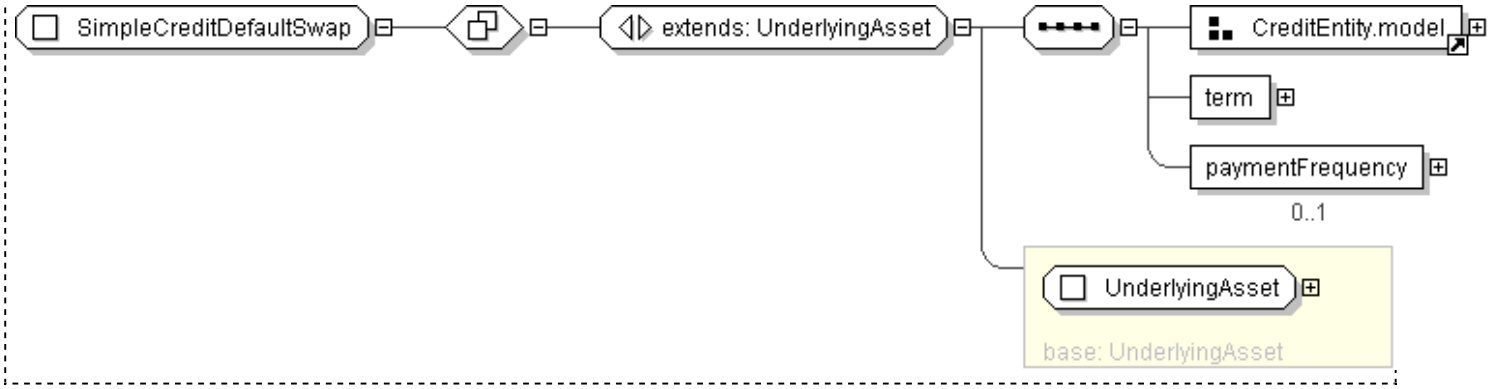
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < SimpleCreditDefaultSwap (by extension)
Sub-types:	None

Name	SimpleCreditDefaultSwap
Used by (from the same schema document)	Element simpleCreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the purposes
    of calculating a contractual payoff. The term \"Exchange\" is assumed to have the
    meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product in
    greater detail. In case of inconsistency between the terms of the simple product
    and those of the detailed definition, the values in the simple product override
    those in the detailed definition.'

   Start Choice [1]
      <referenceEntity> LegalEntity </referenceEntity> [1]
       'The entity for which this is defined.'

      <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
       'An XML reference a credit entity defined elsewhere in the document.'

   End Choice
   <term> Period </term> [1]
    'Specifies the term of the simple CD swap, e.g. 5Y.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the swap pays, e.g. 6M.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SimpleCreditDefaultSwap">
   <xsd:complexContent>
      <xsd:extension base=" UnderlyingAsset ">
         <xsd:sequence>
            <xsd:group ref=" CreditEntity.model "/>
            <xsd:element name="term" type=" Period "/>
            <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SimpleFra

[Table of contents]

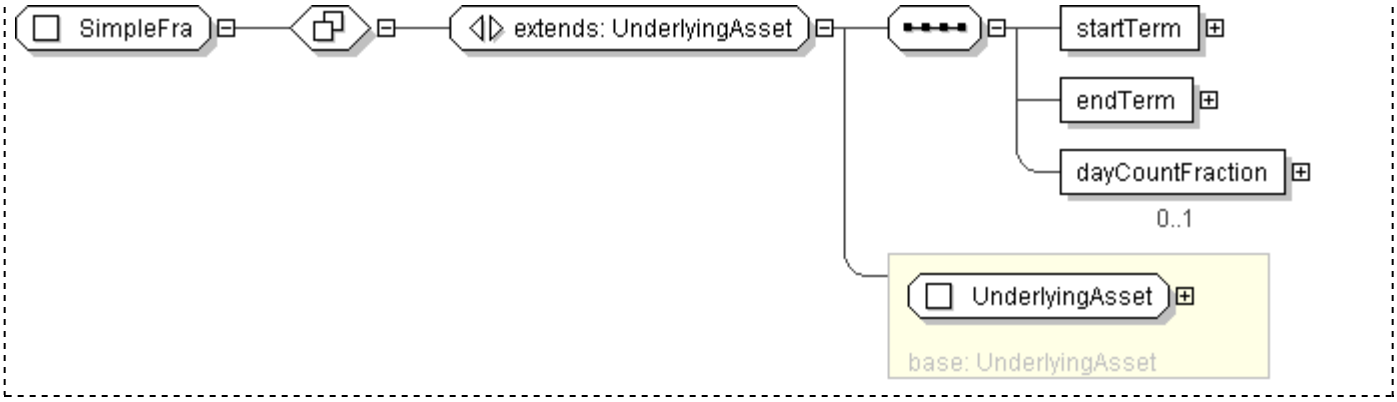
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < SimpleFra (by extension)
Sub-types:	None

Name	SimpleFra
Used by (from the same schema document)	Element simpleFra
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <startTerm> Period </startTerm> [1]
    'Specifies the start term of the simple fra, e.g. 3M.'

   <endTerm> Period </endTerm> [1]
    'Specifies the end term of the simple fra, e.g. 9M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the FRA.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SimpleFra">
  <xsd:complexContent>
    <xsd:extension base=" UnderlyingAsset ">
      <xsd:sequence>
        <xsd:element name="startTerm" type=" Period "/>
        <xsd:element name="endTerm" type=" Period "/>
        <xsd:element name="dayCountFraction" type=" DayCountFraction "
        minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SimpleIRSwap

[Table of contents]

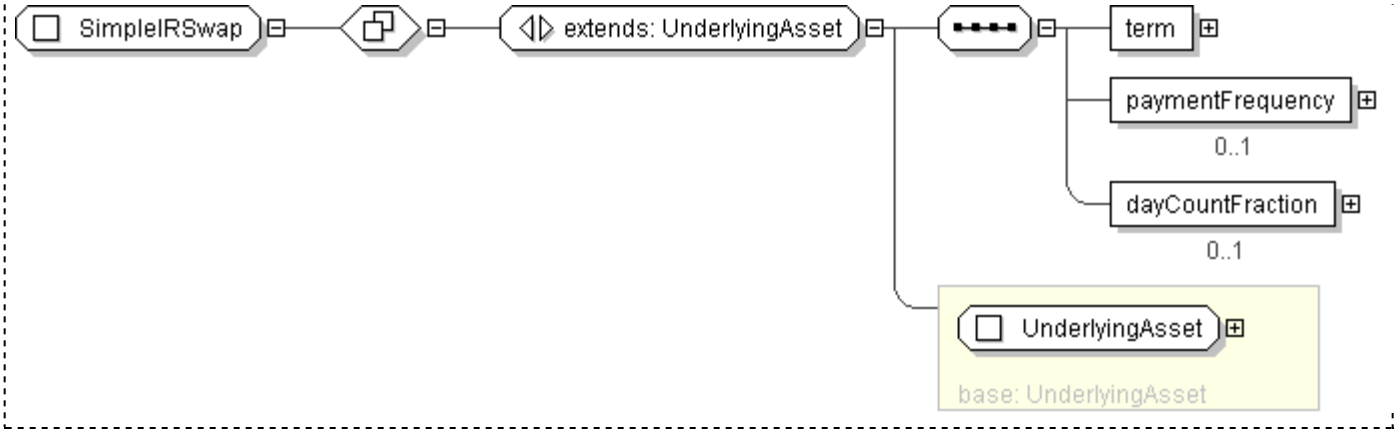
Super-types:	Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension) < SimpleIRSwap (by extension)
Sub-types:	None

Name	SimpleIRSwap
Used by (from the same schema document)	Element simpleIrSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
    'Identification of the underlying asset, using public and/or private
    identifiers.'

   <description> xsd:string </description> [0..1]
    'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
    'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
    'Identification of the exchange on which this asset is transacted for the
    purposes of calculating a contractural payoff. The term \"Exchange\" is assumed
    to have the meaning as defined in the ISDA 2002 Equity Derivatives
    Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
    'Identification of the clearance system associated with the transaction
    exchange.'

   <definition> ProductReference </definition> [0..1]
    'An optional reference to a full FpML product that defines the simple product
    in greater detail. In case of inconsistency between the terms of the simple
    product and those of the detailed definition, the values in the simple product
    override those in the detailed definition.'

   <term> Period </term> [1]
    'Specifies the term of the simple swap, e.g. 5Y.'

   <paymentFrequency> Period </paymentFrequency> [0..1]
    'Specifies the frequency at which the swap pays, e.g. 6M.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
    'The day count basis for the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SimpleIRSwap">
   <xsd:complexContent>
      <xsd:extension base=" UnderlyingAsset ">
         <xsd:sequence>
            <xsd:element name="term" type=" Period "/>
            <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SingleUnderlyer

[Table of contents]

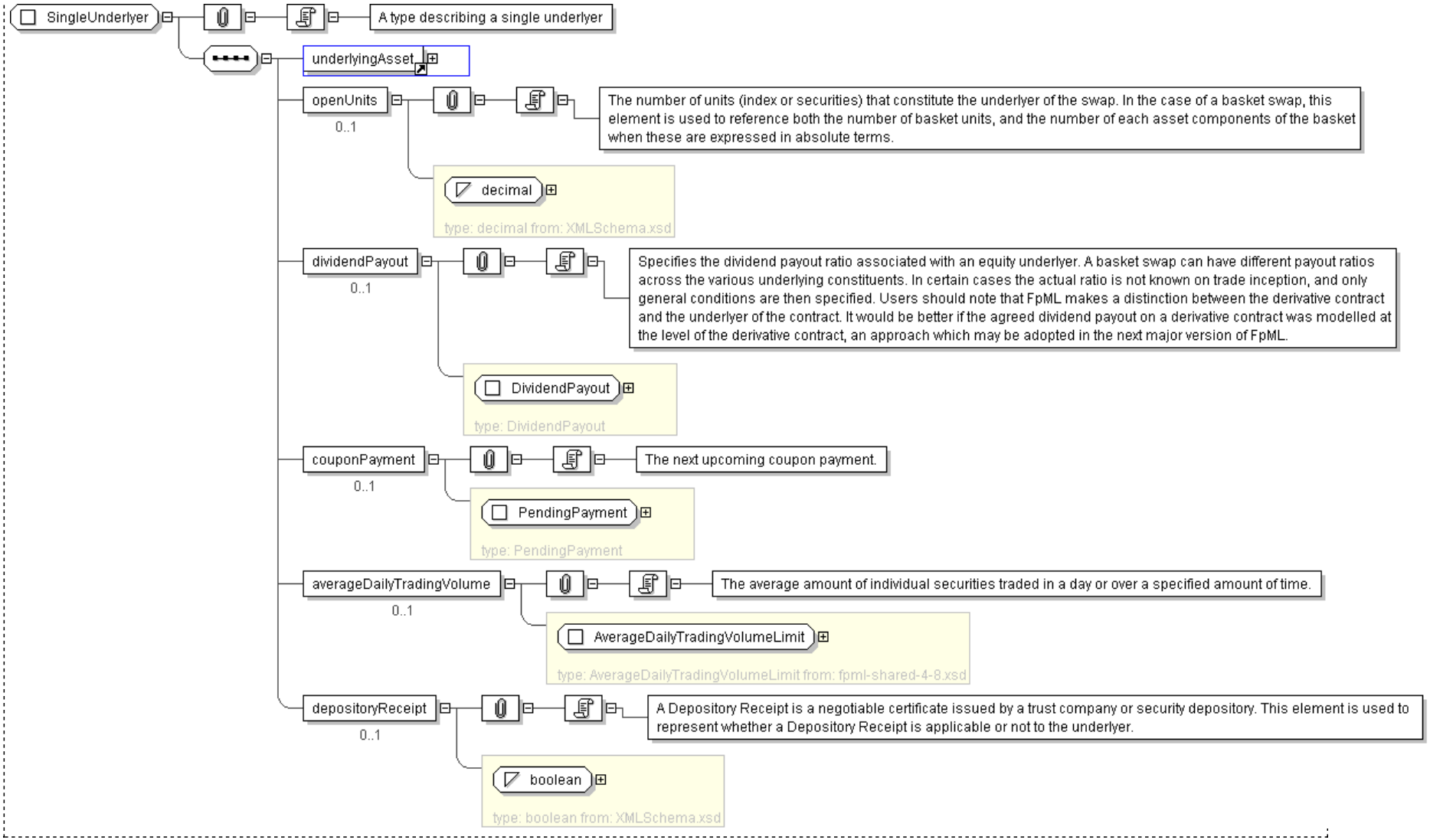
Super-types:	None
Sub-types:	None

Name	SingleUnderlyer
Used by (from the same schema document)	Complex Type Underlyer
Abstract	no
Documentation	A type describing a single underlyer

XML Instance Representation

```
<...>
   <underlyingAsset> ... </underlyingAsset> [1]
   <openUnits> xsd:decimal </openUnits> [0..1]
   'The number of units (index or securities) that constitute the underlyer of the swap. In the case of a basket swap, this element is used
   to reference both the number of basket units, and the number of each asset components of the basket when these are expressed in absolute
   terms.'

   <dividendPayout> DividendPayout </dividendPayout> [0..1]
   'Specifies the dividend payout ratio associated with an equity underlyer. A basket swap can have different payout ratios across the
   various underlying constituents. In certain cases the actual ratio is not known on trade inception, and only general conditions are then
   specified. Users should note that FpML makes a distinction between the derivative contract and the underlyer of the contract. It would
   be better if the agreed dividend payout on a derivative contract was modelled at the level of the derivative contract, an approach which
   may be adopted in the next major version of FpML.'

   <couponPayment> PendingPayment </couponPayment> [0..1]
   'The next upcoming coupon payment.'

   <averageDailyTradingVolume> AverageDailyTradingVolumeLimit </averageDailyTradingVolume> [0..1]
   'The average amount of individual securities traded in a day or over a specified amount of time.'

   <depositoryReceipt> xsd:boolean </depositoryReceipt> [0..1]
   'A Depository Receipt is a negotiable certificate issued by a trust company or security depository. This element is used to represent
   whether a Depository Receipt is applicable or not to the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SingleUnderlyer">
  <xsd:sequence>
    <xsd:element ref=" underlyingAsset "/>
    <xsd:element name="openUnits" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="dividendPayout" type=" DividendPayout " minOccurs="0"/>
    <xsd:element name="couponPayment" type=" PendingPayment " minOccurs="0"/>
    <xsd:element name="averageDailyTradingVolume" type=" AverageDailyTradingVolumeLimit " minOccurs="0"/>
    <xsd:element name="depositoryReceipt" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TimeZone

[Table of contents]

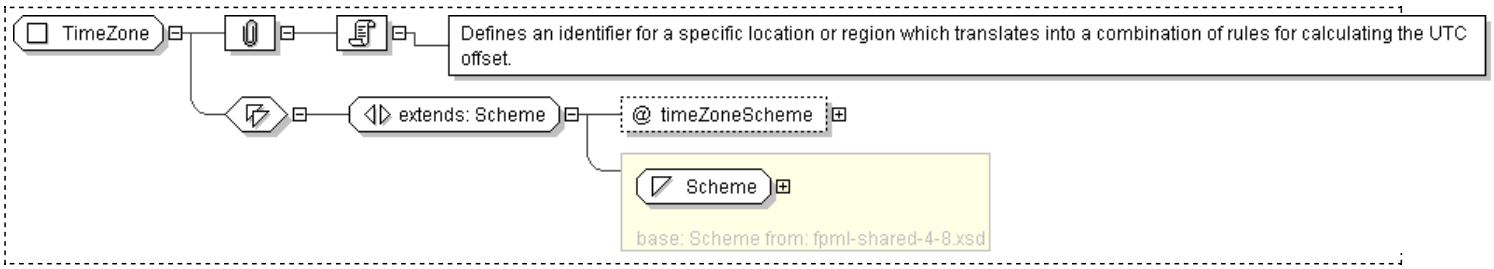
Super-types:	Scheme < TimeZone (by extension)
Sub-types:	None

Name	TimeZone
Used by (from the same schema document)	Complex Type CommodityBusinessCalendarTime
Abstract	no
Documentation	Defines an identifier for a specific location or region which translates into a combination of rules for calculating the UTC offset.

XML Instance Representation

```
<...
 timeZoneScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TimeZone">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="timeZoneScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Underlyer

[Table of contents]

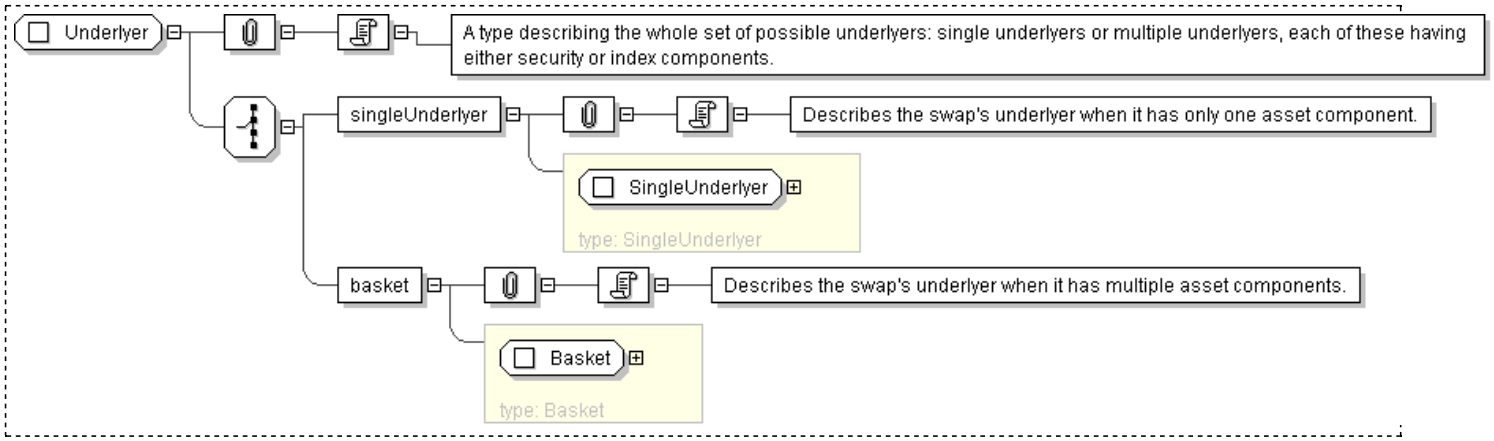
Super-types:	None
Sub-types:	None

Name	Underlyer
Abstract	no
Documentation	A type describing the whole set of possible underlyers: single underlyers or multiple underlyers, each of these having either security or index components.

XML Instance Representation

```
<...>
  Start Choice [1]
    <singleUnderlyer> SingleUnderlyer </singleUnderlyer> [1]
     'Describes the swap\'s underlyer when it has only one asset component.'

    <basket> Basket </basket> [1]
     'Describes the swap\'s underlyer when it has multiple asset components.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Underlyer">
  <xsd:choice>
    <xsd:element name="singleUnderlyer" type=" SingleUnderlyer "/>
    <xsd:element name="basket" type=" Basket "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: UnderlyingAsset

[Table of contents]

Super-types: Asset < IdentifiedAsset (by extension) < UnderlyingAsset (by extension)

Sub-types:

- Deposit (by extension)
- ExchangeTraded (by extension)
  - Bond (by extension)
    - ConvertibleBond (by extension)
  - EquityAsset (by extension)
  - ExchangeTradedCalculatedPrice (by extension)
    - ExchangeTradedFund (by extension)
    - Index (by extension)
  - ExchangeTradedContract (by extension)
  - Future (by extension)
- FxRateAsset (by extension)
- Loan (by extension)
- Mortgage (by extension)
- MutualFund (by extension)
- RateIndex (by extension)
- SimpleCreditDefaultSwap (by extension)
- SimpleFra (by extension)
- SimpleIRSwap (by extension)

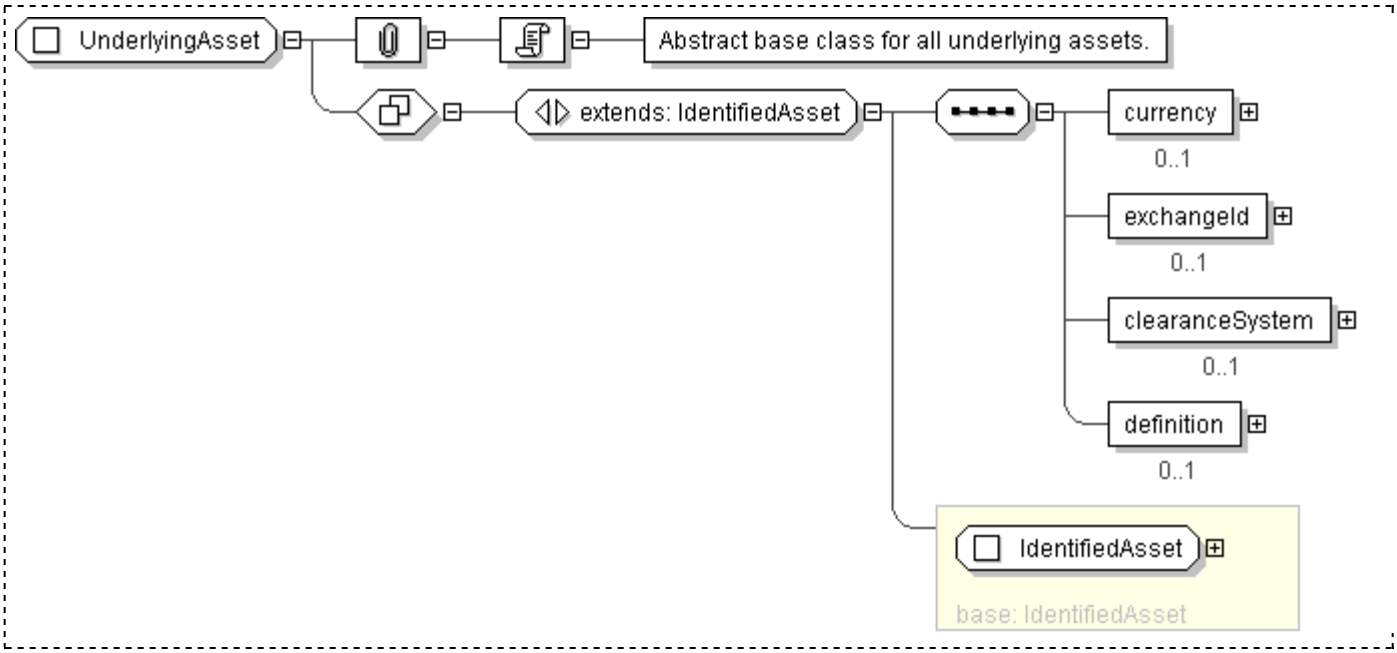
Name	UnderlyingAsset
Abstract	yes
Documentation	Abstract base class for all underlying assets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private
   identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   <currency> IdentifiedCurrency </currency> [0..1]
   'Trading currency of the underlyer when transacted as a cash instrument.'

   <exchangeId> ExchangeId </exchangeId> [0..1]
   'Identification of the exchange on which this asset is transacted for the
   purposes of calculating a contractual payoff. The term \"Exchange\" is assumed
   to have the meaning as defined in the ISDA 2002 Equity Derivatives
   Definitions.'

   <clearanceSystem> ClearanceSystem </clearanceSystem> [0..1]
   'Identification of the clearance system associated with the transaction
   exchange.'

   <definition> ProductReference </definition> [0..1]
   'An optional reference to a full FpML product that defines the simple product
   in greater detail. In case of inconsistency between the terms of the simple
   product and those of the detailed definition, the values in the simple product
```

```
   override those in the detailed definition.'
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="UnderlyingAsset" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" IdentifiedAsset ">
      <xsd:sequence>
        <xsd:element name="currency" type=" IdentifiedCurrency " minOccurs="0"/>
        <xsd:element name="exchangeId" type=" ExchangeId " minOccurs="0"/>
        <xsd:element name="clearanceSystem" type=" ClearanceSystem " minOccurs="0"/>
        <xsd:element name="definition" type=" ProductReference " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: UnderlyingAssetTranche

[Table of contents]

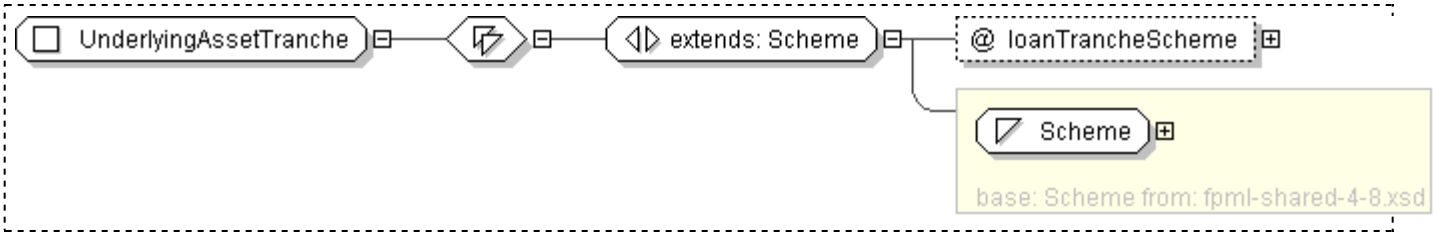
Super-types:	Scheme < UnderlyingAssetTranche (by extension)
Sub-types:	None

Name	UnderlyingAssetTranche
Used by (from the same schema document)	Complex Type Loan
Abstract	no

XML Instance Representation

```
<...
 loanTrancheScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="UnderlyingAssetTranche">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="loanTrancheScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/underlying-asset-tranche"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: bondOption
- Global Definitions
  - Complex Type: BondOption
  - Complex Type: BondOptionStrike
  - Complex Type: MakeWholeAmount
  - Complex Type: ReferenceSwapCurve
  - Complex Type: SwapCurveValuation
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ○ fpml-option-shared-4-8.xsd
	  ○ fpml-mktenv-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-option-shared-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-mktenv-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
```

```
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

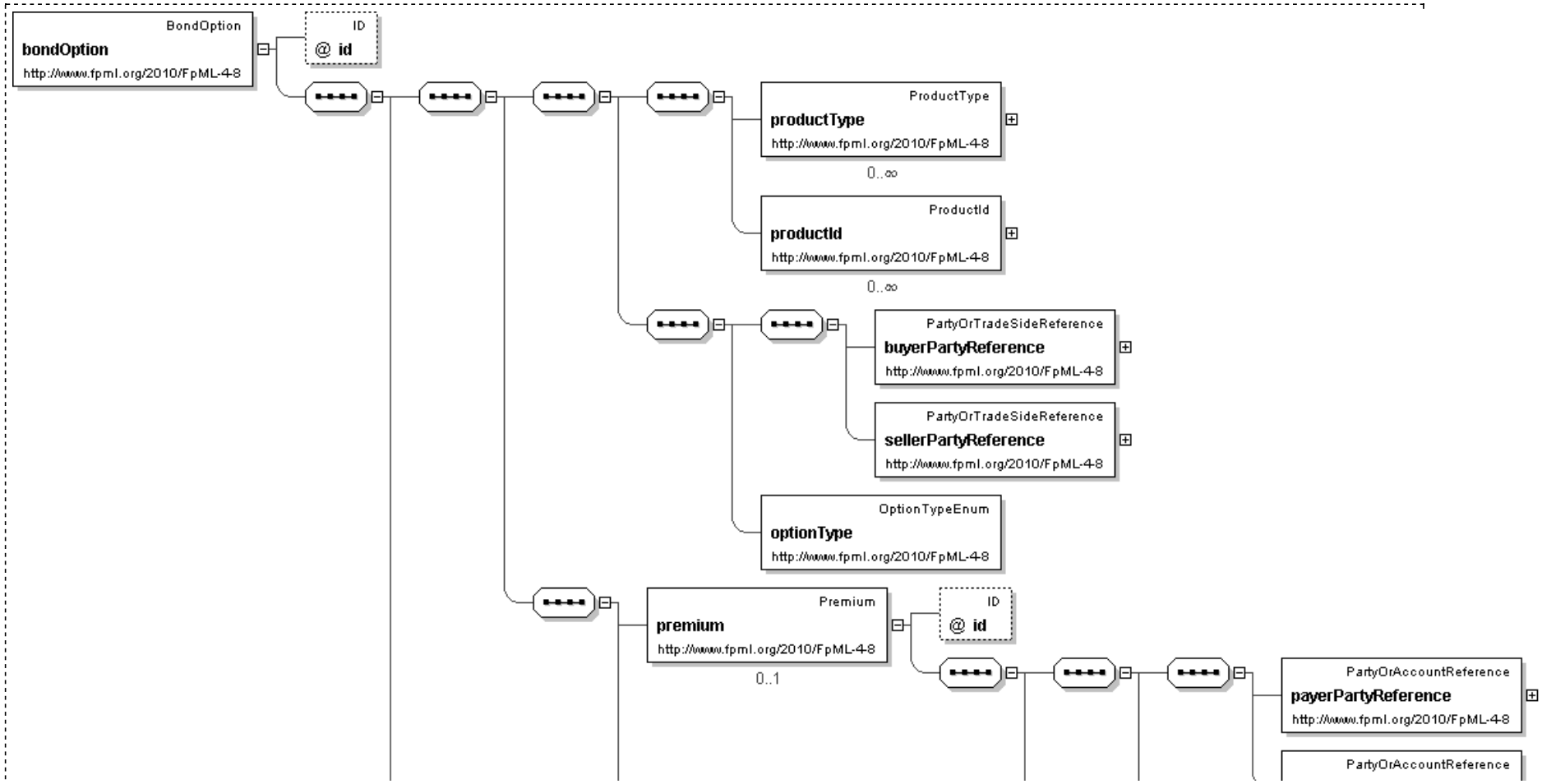
Element: bondOption

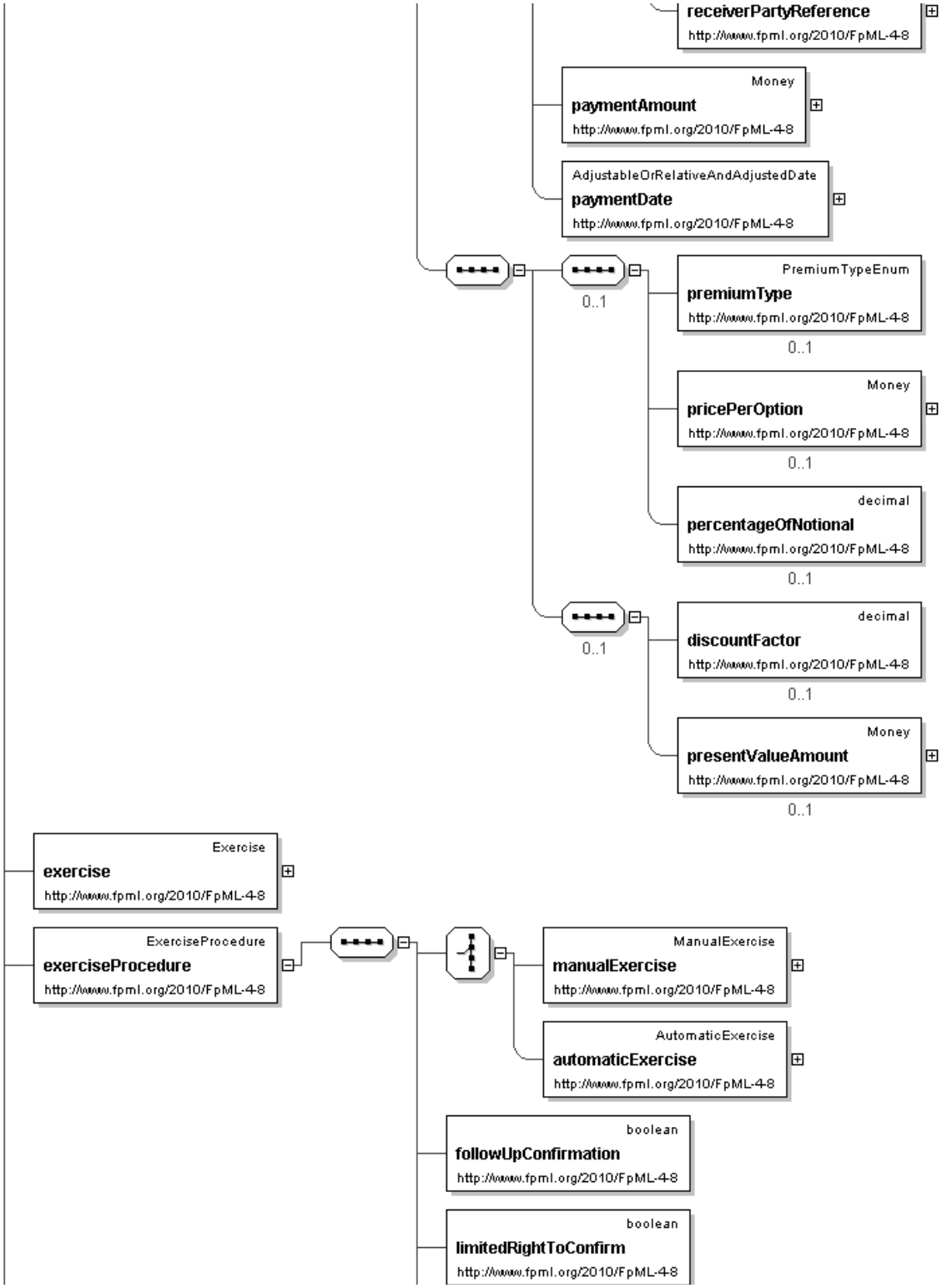
[Table of contents]

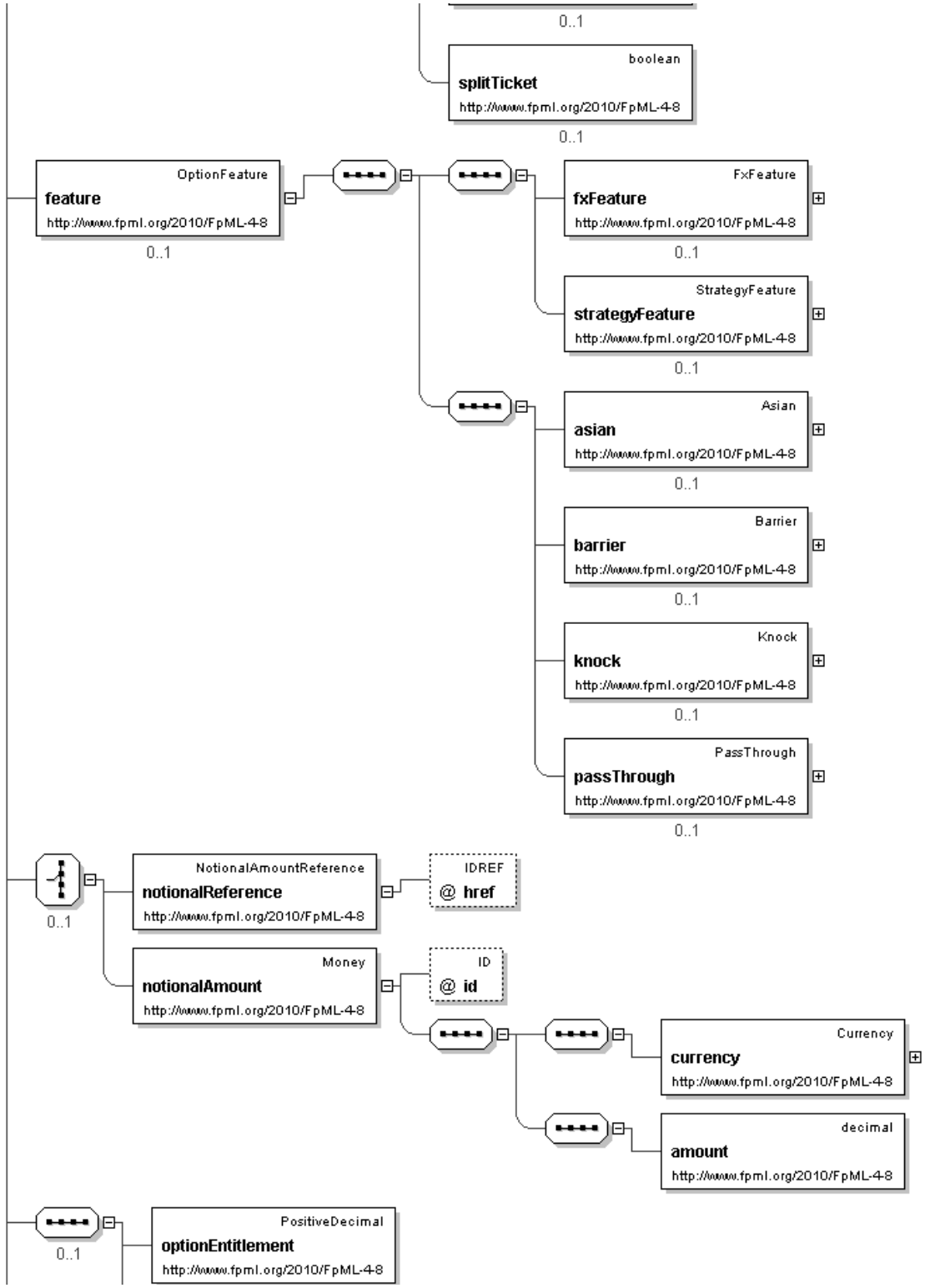
- *This element can be used wherever the following element is referenced:*
  - product

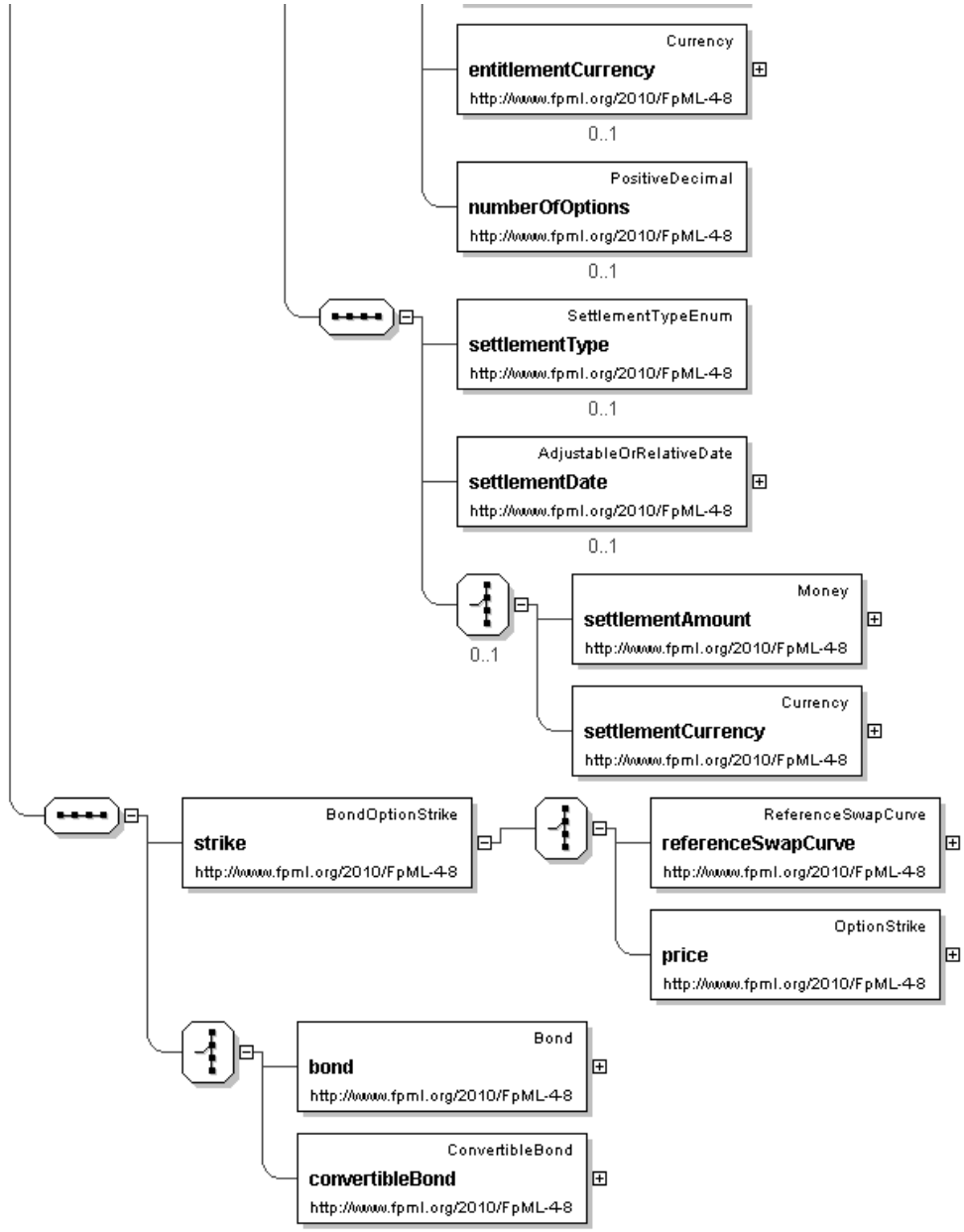
Name	bondOption
Type	BondOption
Nillable	no
Abstract	no
Documentation	A component describing a Bond Option product.

Logical Diagram




receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeAndAdjustedDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumTypeEnum
premiumType
http://www.fpml.org/2010/FpML-4-8
0..1
Money
pricePerOption
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
percentageOfNotional
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Exercise
exercise
http://www.fpml.org/2010/FpML-4-8
ExerciseProcedure
exerciseProcedure
http://www.fpml.org/2010/FpML-4-8
ManualExercise
manualExercise
http://www.fpml.org/2010/FpML-4-8
AutomaticExercise
automaticExercise
http://www.fpml.org/2010/FpML-4-8
boolean
followUpConfirmation
http://www.fpml.org/2010/FpML-4-8
boolean
limitedRightToConfirm
http://www.fpml.org/2010/FpML-4-8


0..1
boolean
splitTicket
http://www.fpml.org/2010/FpML-4-8
0..1
OptionFeature
feature
http://www.fpml.org/2010/FpML-4-8
0..1
FxFeature
fxFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrategyFeature
strategyFeature
http://www.fpml.org/2010/FpML-4-8
0..1
Asian
asian
http://www.fpml.org/2010/FpML-4-8
0..1
Barrier
barrier
http://www.fpml.org/2010/FpML-4-8
0..1
Knock
knock
http://www.fpml.org/2010/FpML-4-8
0..1
PassThrough
passThrough
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
NotionalAmountReference
notionalReference
http://www.fpml.org/2010/FpML-4-8
IDREF
@ href
Money
notionalAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
0..1
PositiveDecimal
optionEntitlement
http://www.fpml.org/2010/FpML-4-8

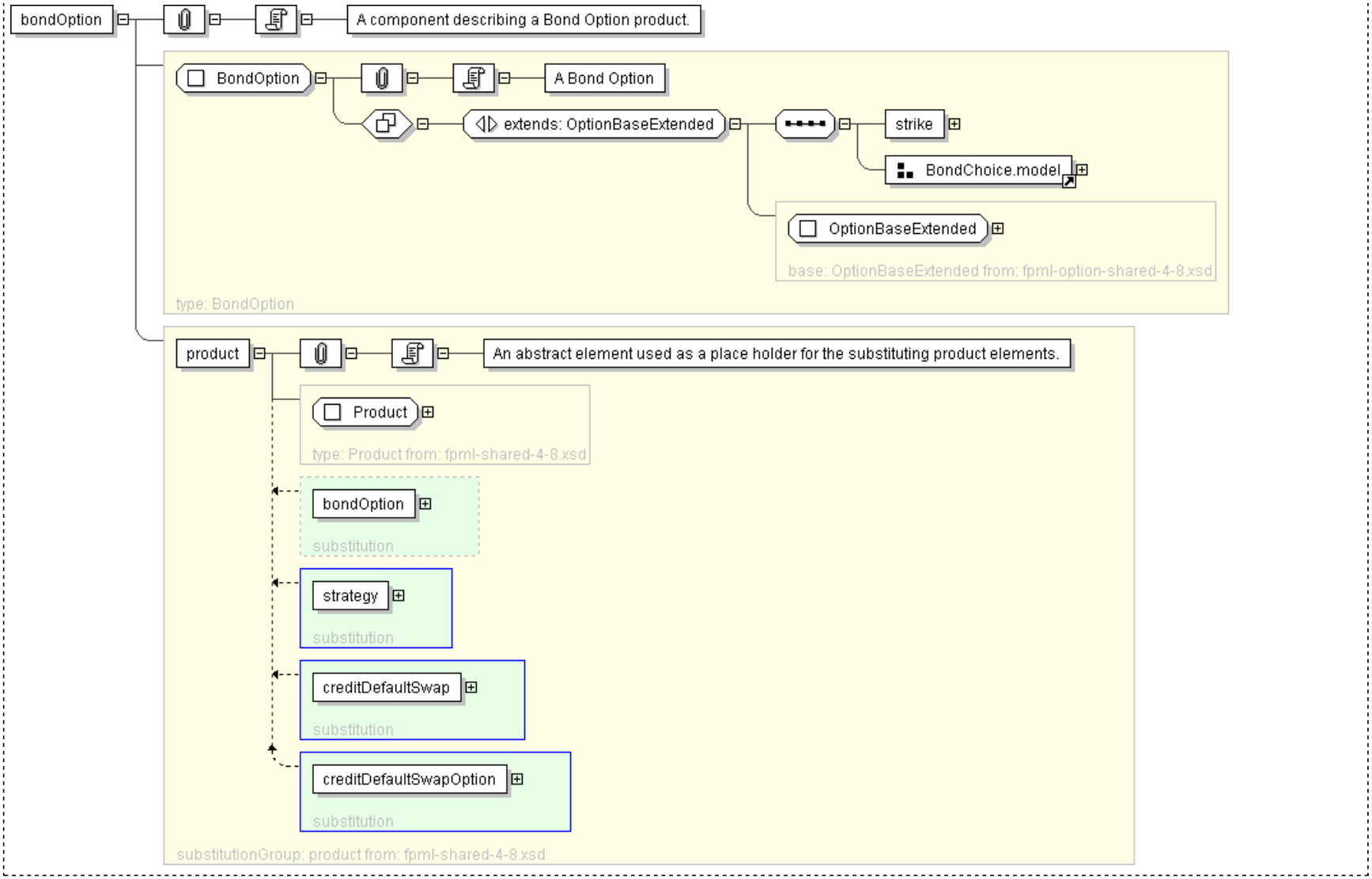


XML Instance Representation

```
<bondOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
   that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
```

```
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See
   2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and
   in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default option type, while payer/receiver indicator
   is used for options index credit default swaps, consistently with the industry practice. Straddle is used for the case of
   straddle strategy, that combine a call and a put with the same strike.'

   <premium> Premium </premium> [0..1]
   'The option premium payable by the buyer to the seller.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [1]
   'A set of parameters defining procedures associated with the exercise.'

   <feature> OptionFeature </feature> [0..1]
   'An Option feature such as quanto, asian, barrier, knock.'

   Start Choice [0..1]
'A choice between an explicit representation of the notional amount, or a reference to a notional amount defined elsewhere in this
document.'


      <notionalReference> NotionalAmountReference </notionalReference> [1]
      <notionalAmount> Money </notionalAmount> [1]
   End Choice
   Start Group: OptionDenomination.model [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
      'The number of units of underlyer per option comprised in the option transaction.'

      <entitlementCurrency> Currency </entitlementCurrency> [0..1]
      'TODO'

      <numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
      'The number of options comprised in the option transaction.'

   End Group: OptionDenomination.model
   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
         'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
         'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <strike> BondOptionStrike </strike> [1]
   'Strike of the the Bond Option.'

   Start Choice [1]
      <bond> ... </bond> [1]
      'A bond instrument referenced by a contract'

      <convertibleBond> ... </convertibleBond> [1]
      'A convertible bond instrument referenced by a contract.'

   End Choice
</bondOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="bondOption" type=" BondOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BondOption

[Table of contents]

Super-types:	OptionBaseExtended < BondOption (by extension)
Sub-types:	None

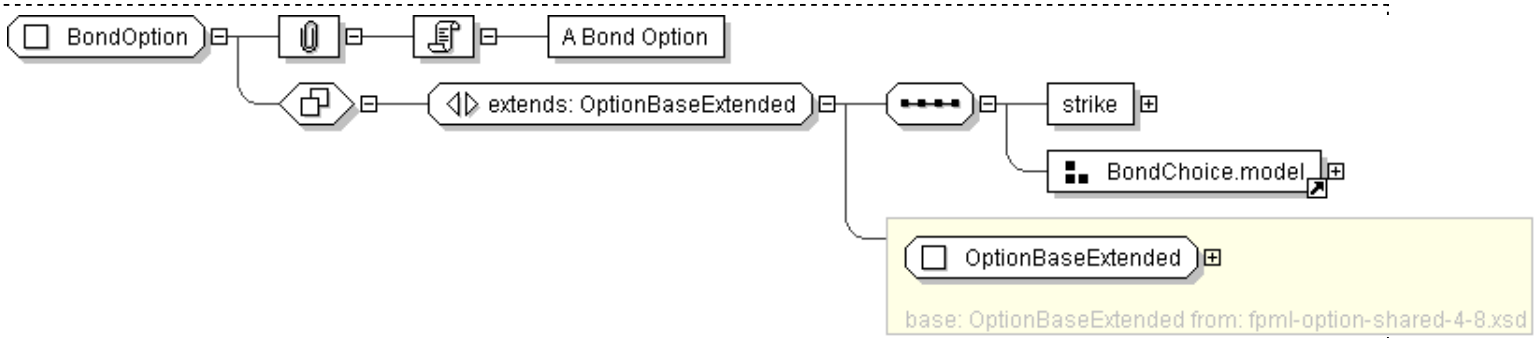
Name	BondOption
Used by (from the same schema document)	Element bondOption
Abstract	no
Documentation	A Bond Option

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization
   using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the
   case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants
   the rights defined by this instrument and in return receives a payment for it. See
   2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default
   option type, while payer/receiver indicator is used for options index credit default
   swaps, consistently with the industry practice. Straddle is used for the case of
   straddle strategy, that combine a call and a put with the same strike.'

   <premium> Premium </premium> [0..1]
   'The option premium payable by the buyer to the seller.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [1]
   'A set of parameters defining procedures associated with the exercise.'

   <feature> OptionFeature </feature> [0..1]
   'An Option feature such as quanto, asian, barrier, knock.'

   Start Choice [0..1]
'A choice between an explicit representation of the notional amount, or a reference to a
notional amount defined elsewhere in this document.'


      <notionalReference> NotionalAmountReference </notionalReference> [1]
      <notionalAmount> Money </notionalAmount> [1]
   End Choice
   Start Group: OptionDenomination.model [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
      'The number of units of underlyer per option comprised in the option transaction.'

      <entitlementCurrency> Currency </entitlementCurrency> [0..1]
      'TODO'
```

```
      <numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
      'The number of options comprised in the option transaction.'

   End Group: OptionDenomination.model
   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
         'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
         'Settlement Currency for use where the Settlement Amount cannot be known in
         advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <strike> BondOptionStrike </strike> [1]
   'Strike of the the Bond Option.'

   Start Choice [1]
      <bond> ... </bond> [1]
      'A bond instrument referenced by a contract'

      <convertibleBond> ... </convertibleBond> [1]
      'A convertible bond instrument referenced by a contract.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BondOption">
   <xsd:complexContent>
      <xsd:extension base=" OptionBaseExtended ">
         <xsd:sequence>
            <xsd:element name="strike" type=" BondOptionStrike "/>
            <xsd:group ref=" BondChoice.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BondOptionStrike

[Table of contents]

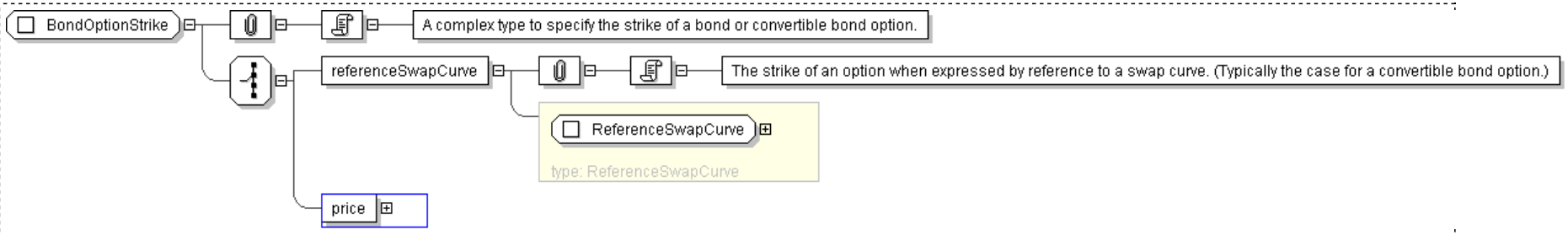
Super-types:	None
Sub-types:	None

Name	BondOptionStrike
Used by (from the same schema document)	Complex Type BondOption
Abstract	no
Documentation	A complex type to specify the strike of a bond or convertible bond option.

XML Instance Representation

```
<...>
  Start Choice [1]
    <referenceSwapCurve> ReferenceSwapCurve </referenceSwapCurve> [1]
    'The strike of an option when expressed by reference to a swap curve. (Typically the case for a convertible bond option.)'

    <price> OptionStrike </price> [1]
  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BondOptionStrike">
  <xsd:choice>
    <xsd:element name="referenceSwapCurve" type=" ReferenceSwapCurve "/>
    <xsd:element name="price" type=" OptionStrike "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MakeWholeAmount

[Table of contents]

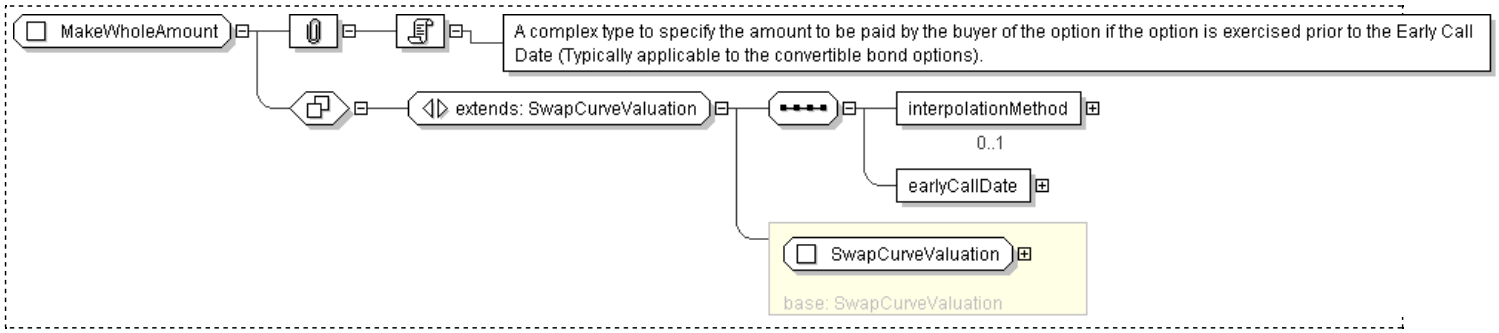
Super-types:	SwapCurveValuation < MakeWholeAmount (by extension)
Sub-types:	None

Name	MakeWholeAmount
Used by (from the same schema document)	Complex Type ReferenceSwapCurve
Abstract	no
Documentation	A complex type to specify the amount to be paid by the buyer of the option if the option is exercised prior to the Early Call Date (Typically applicable to the convertible bond options).

XML Instance Representation

```
<...>
  <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
  <indexTenor> Period </indexTenor> [0..1]
  'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

  <spread> xsd:decimal </spread> [1]
  'Spread in basis points over the floating rate index.'

  <side> QuotationSideEnum </side> [0..1]
  'The side (bid/mid/ask) of the measure.'

  <interpolationMethod> InterpolationMethod </interpolationMethod> [0..1]
  'The type of interpolation method that the calculation agent reserves the right to use.'

  <earlyCallDate> IdentifiedDate </earlyCallDate> [1]
  'Date prior to which the option buyer will have to pay a Make Whole Amount to the option seller if he/she
  exercises the option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MakeWholeAmount">
  <xsd:complexContent>
    <xsd:extension base=" SwapCurveValuation ">
      <xsd:sequence>
        <xsd:element name="interpolationMethod" type=" InterpolationMethod " minOccurs="0"/>
        <xsd:element name="earlyCallDate" type=" IdentifiedDate "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceSwapCurve

[Table of contents]

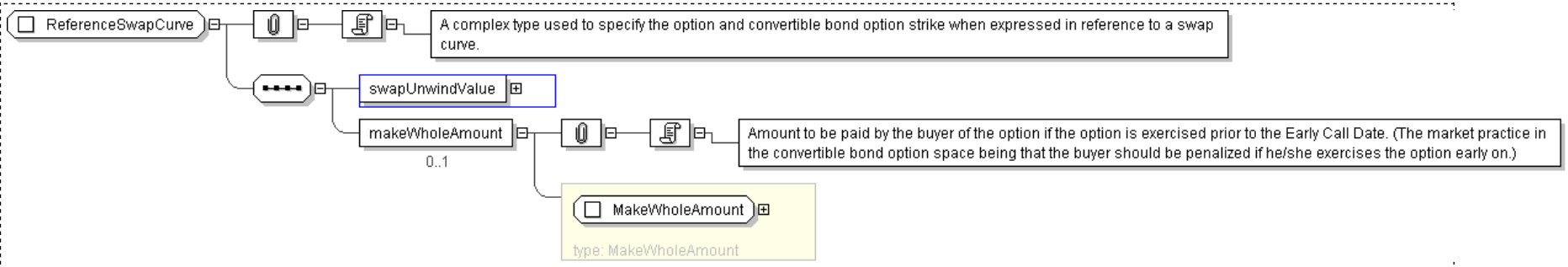
Super-types:	None
Sub-types:	None

Name	ReferenceSwapCurve
Used by (from the same schema document)	Complex Type BondOptionStrike
Abstract	no
Documentation	A complex type used to specify the option and convertible bond option strike when expressed in reference to a swap curve.

XML Instance Representation

```
<...>
   <swapUnwindValue> SwapCurveValuation </swapUnwindValue> [1]
   <makeWholeAmount> MakeWholeAmount </makeWholeAmount> [0..1]
   'Amount to be paid by the buyer of the option if the option is exercised prior to the Early Call Date. (The market practice in the
   convertible bond option space being that the buyer should be penalized if he/she exercises the option early on.)'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferenceSwapCurve">
   <xsd:sequence>
      <xsd:element name="swapUnwindValue" type=" SwapCurveValuation "/>
      <xsd:element name="makeWholeAmount" type=" MakeWholeAmount " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SwapCurveValuation

[Table of contents]

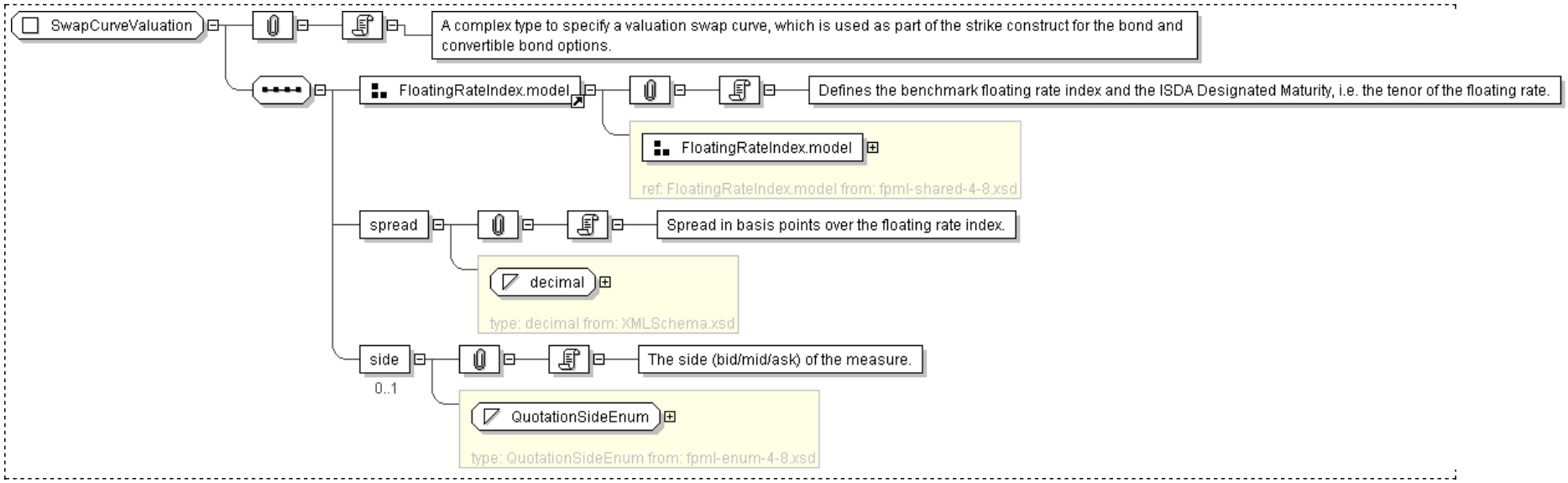
Super-types:	None
Sub-types:	• MakeWholeAmount (by extension)

Name	SwapCurveValuation
Used by (from the same schema document)	Complex Type ReferenceSwapCurve
Abstract	no
Documentation	A complex type to specify a valuation swap curve, which is used as part of the strike construct for the bond and convertible bond options.

XML Instance Representation

```
<...>
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <spread> xsd:decimal </spread> [1]
   'Spread in basis points over the floating rate index.'

   <side> QuotationSideEnum </side> [0..1]
   'The side (bid/mid/ask) of the measure.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SwapCurveValuation">
   <xsd:sequence>
      <xsd:group ref=" FloatingRateIndex.model "/>
```

```
      <xsd:element name="spread" type=" xsd:decimal "/>
      <xsd:element name="side" type=" QuotationSideEnum " minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: creditDefaultSwap
  - Element: creditDefaultSwapOption
- Global Definitions
  - Complex Type: AdditionalFixedPayments
  - Complex Type: AdditionalTerm
  - Complex Type: AdjustedPaymentDates
  - Complex Type: BasketReferenceInformation
  - Complex Type: CalculationAmount
  - Complex Type: CashSettlementTerms
  - Complex Type: CreditDefaultSwap
  - Complex Type: CreditDefaultSwapOption
  - Complex Type: CreditOptionStrike
  - Complex Type: DeliverableObligations
  - Complex Type: DeprecatedScheduledTerminationDate
  - Complex Type: EntityType
  - Complex Type: FeeLeg
  - Complex Type: FixedAmountCalculation
  - Complex Type: FixedRate
  - Complex Type: FixedRateReference
  - Complex Type: FloatingAmountEvents
  - Complex Type: FloatingAmountProvisions
  - Complex Type: GeneralTerms
  - Complex Type: IndexAnnexSource
  - Complex Type: IndexId
  - Complex Type: IndexName
  - Complex Type: IndexReferenceInformation
  - Complex Type: InitialPayment
  - Complex Type: InterestShortFall
  - Complex Type: LoanParticipation
  - Complex Type: MatrixSource
  - Complex Type: MultipleValuationDates
  - Complex Type: NotDomesticCurrency
  - Complex Type: Obligations
  - Complex Type: PCDeliverableObligationCharac
  - Complex Type: PeriodicPayment
  - Complex Type: PhysicalSettlementPeriod
  - Complex Type: PhysicalSettlementTerms
  - Complex Type: ProtectionTerms
  - Complex Type: ProtectionTermsReference
  - Complex Type: ReferenceInformation
  - Complex Type: ReferenceObligation
  - Complex Type: ReferencePair
  - Complex Type: ReferencePool
  - Complex Type: ReferencePoolItem
  - Complex Type: ScheduledTerminationDate
  - Complex Type: SettledEntityMatrix
  - Complex Type: SettlementTerms
  - Complex Type: SettlementTermsReference
  - Complex Type: SinglePayment
  - Complex Type: SingleValuationDate
  - Complex Type: SpecifiedCurrency
  - Complex Type: Tranche

- Complex Type: ValuationDate
- Model Group: FixedRecovery.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-option-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-option-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-

all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

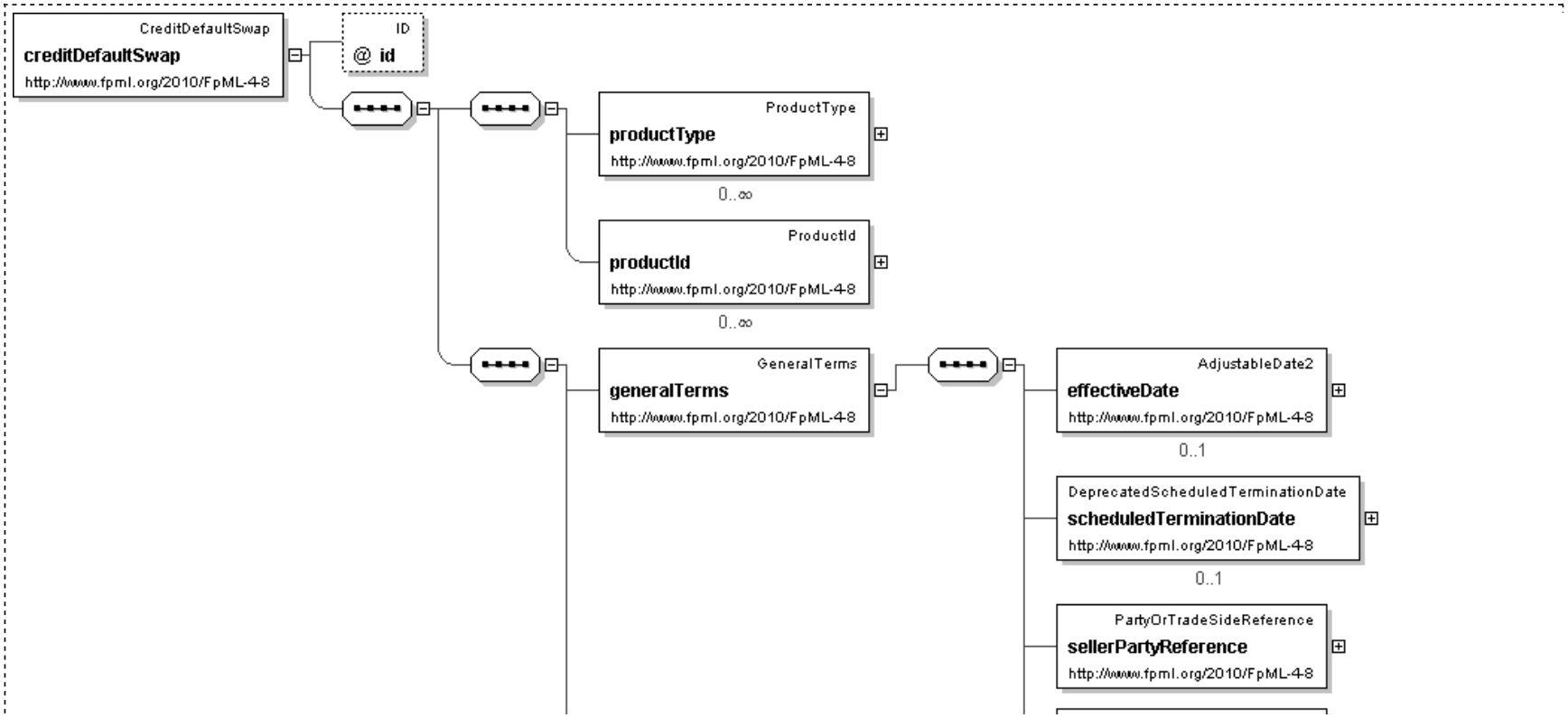
Element: creditDefaultSwap

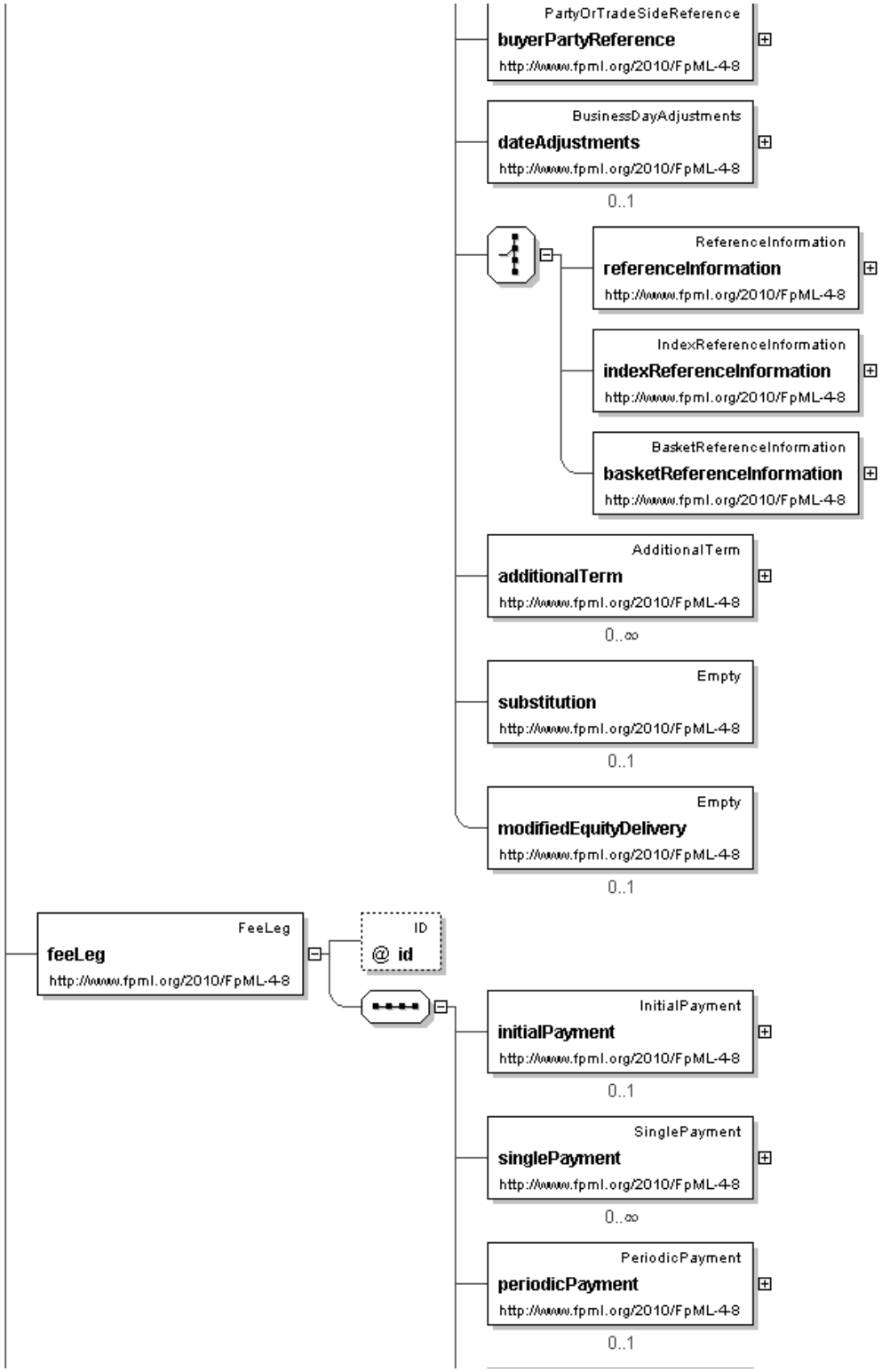
[Table of contents]

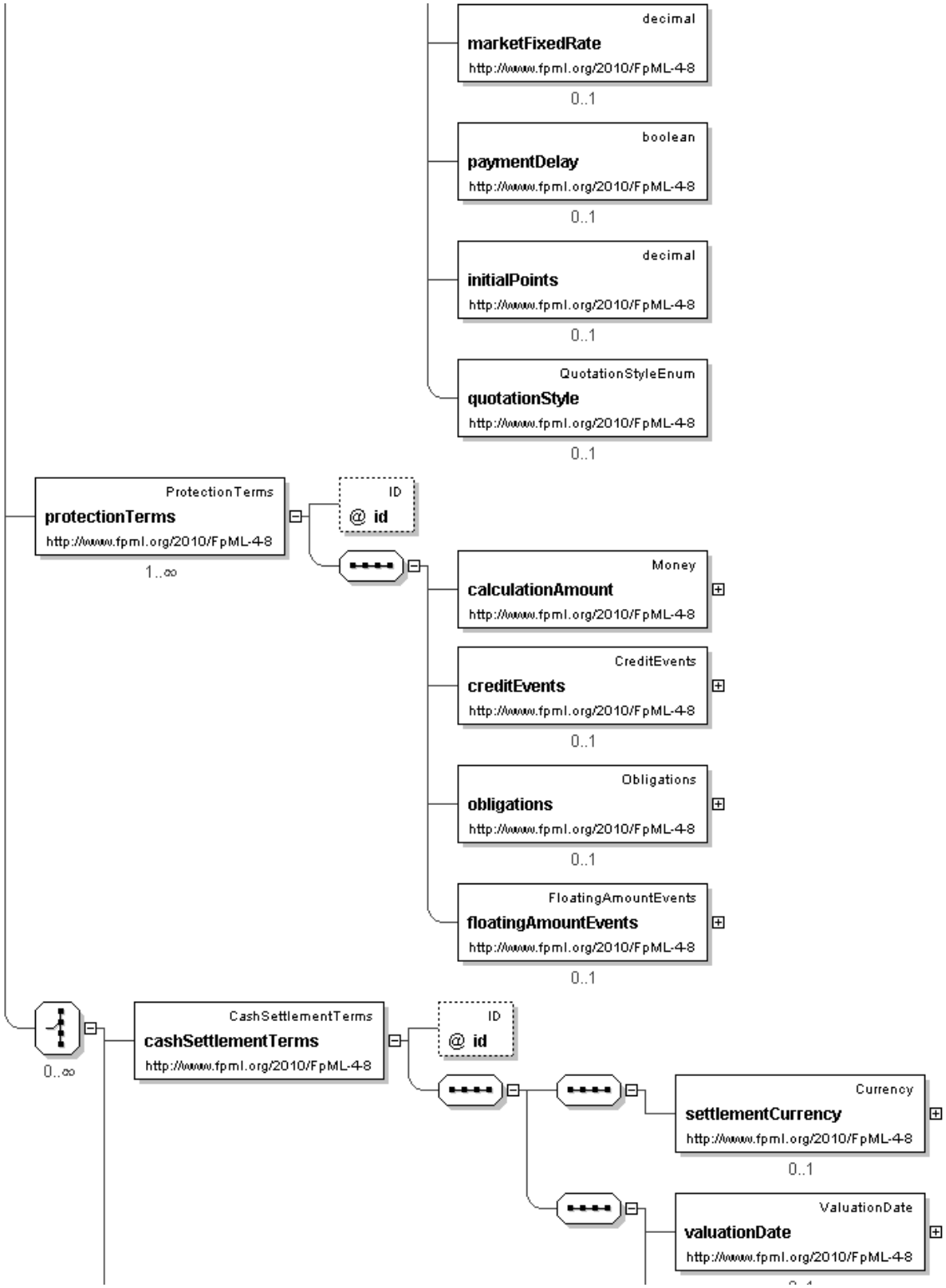
- *This element can be used wherever the following element is referenced:*
  - product

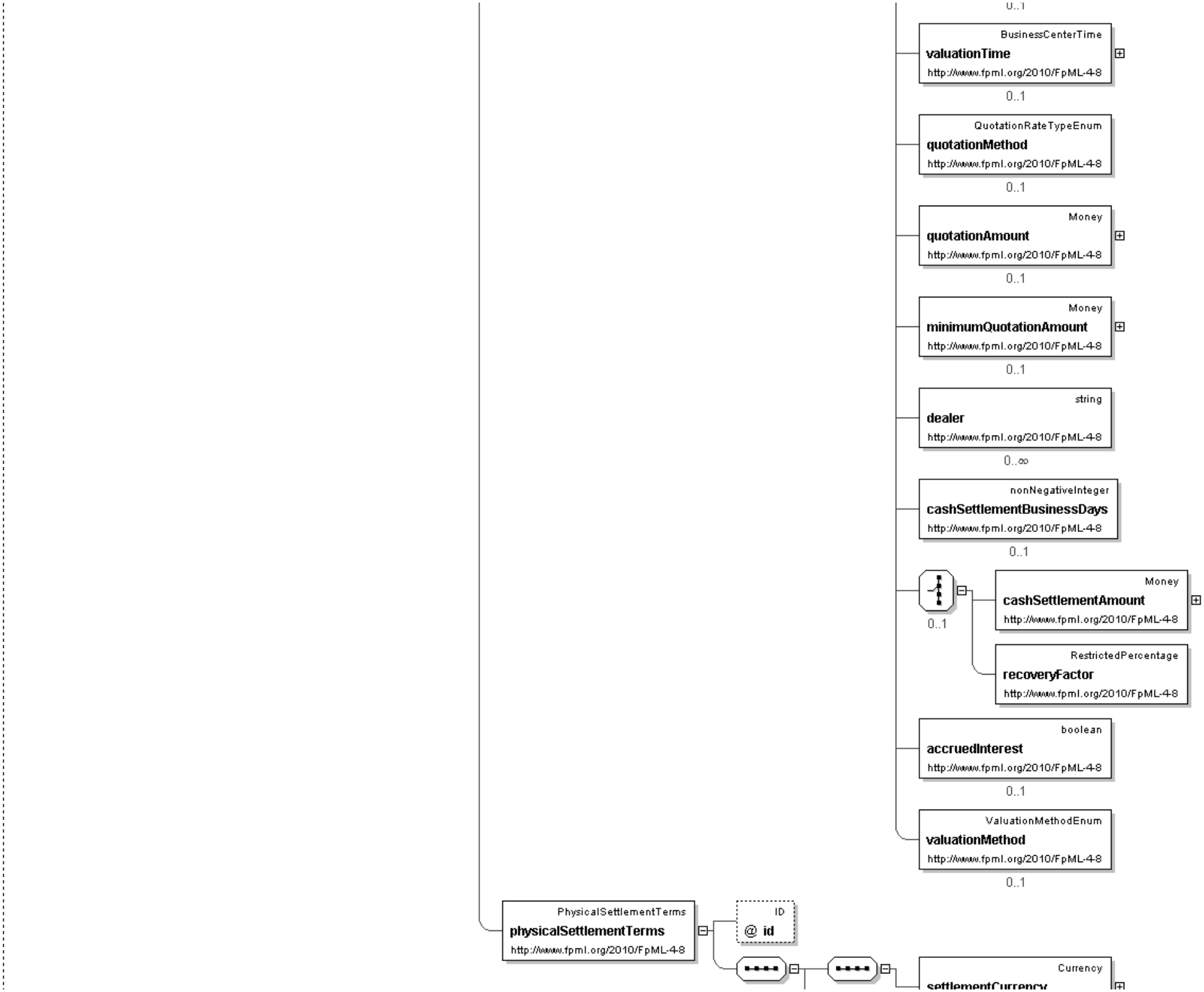
Name	creditDefaultSwap
Used by (from the same schema document)	Complex Type CreditDefaultSwapOption
Type	CreditDefaultSwap
Nillable	no
Abstract	no
Documentation	In a credit default swap one party (the protection seller) agrees to compensate another party (the protection buyer) if a specified company or Sovereign (the reference entity) experiences a credit event, indicating it is or may be unable to service its debts. The protection seller is typically paid a fee and/or premium, expressed as an annualized percent of the notional in basis points, regularly over the life of the transaction or otherwise as agreed by the parties.

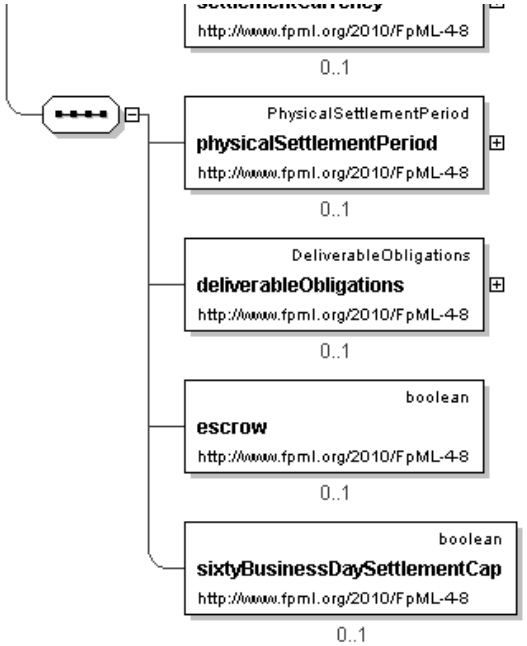
Logical Diagram




PartyOrTradeSideReference
buyerPartyReference
http://www.fpml.org/2010/FpML-4-8
BusinessDayAdjustments
dateAdjustments
http://www.fpml.org/2010/FpML-4-8
0..1
ReferenceInformation
referenceInformation
http://www.fpml.org/2010/FpML-4-8
IndexReferenceInformation
indexReferenceInformation
http://www.fpml.org/2010/FpML-4-8
BasketReferenceInformation
basketReferenceInformation
http://www.fpml.org/2010/FpML-4-8
AdditionalTerm
additionalTerm
http://www.fpml.org/2010/FpML-4-8
0..∞
Empty
substitution
http://www.fpml.org/2010/FpML-4-8
0..1
Empty
modifiedEquityDelivery
http://www.fpml.org/2010/FpML-4-8
0..1
FeeLeg
feeLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
InitialPayment
initialPayment
http://www.fpml.org/2010/FpML-4-8
0..1
SinglePayment
singlePayment
http://www.fpml.org/2010/FpML-4-8
0..∞
PeriodicPayment
periodicPayment
http://www.fpml.org/2010/FpML-4-8
0..1


decimal
marketFixedRate
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
paymentDelay
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
initialPoints
http://www.fpml.org/2010/FpML-4-8
0..1
QuotationStyleEnum
quotationStyle
http://www.fpml.org/2010/FpML-4-8
0..1
ProtectionTerms
protectionTerms
http://www.fpml.org/2010/FpML-4-8
1..∞
ID
@ id
Money
calculationAmount
http://www.fpml.org/2010/FpML-4-8
CreditEvents
creditEvents
http://www.fpml.org/2010/FpML-4-8
0..1
Obligations
obligations
http://www.fpml.org/2010/FpML-4-8
0..1
FloatingAmountEvents
floatingAmountEvents
http://www.fpml.org/2010/FpML-4-8
0..1
0..∞
CashSettlementTerms
cashSettlementTerms
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
ValuationDate
valuationDate
http://www.fpml.org/2010/FpML-4-8

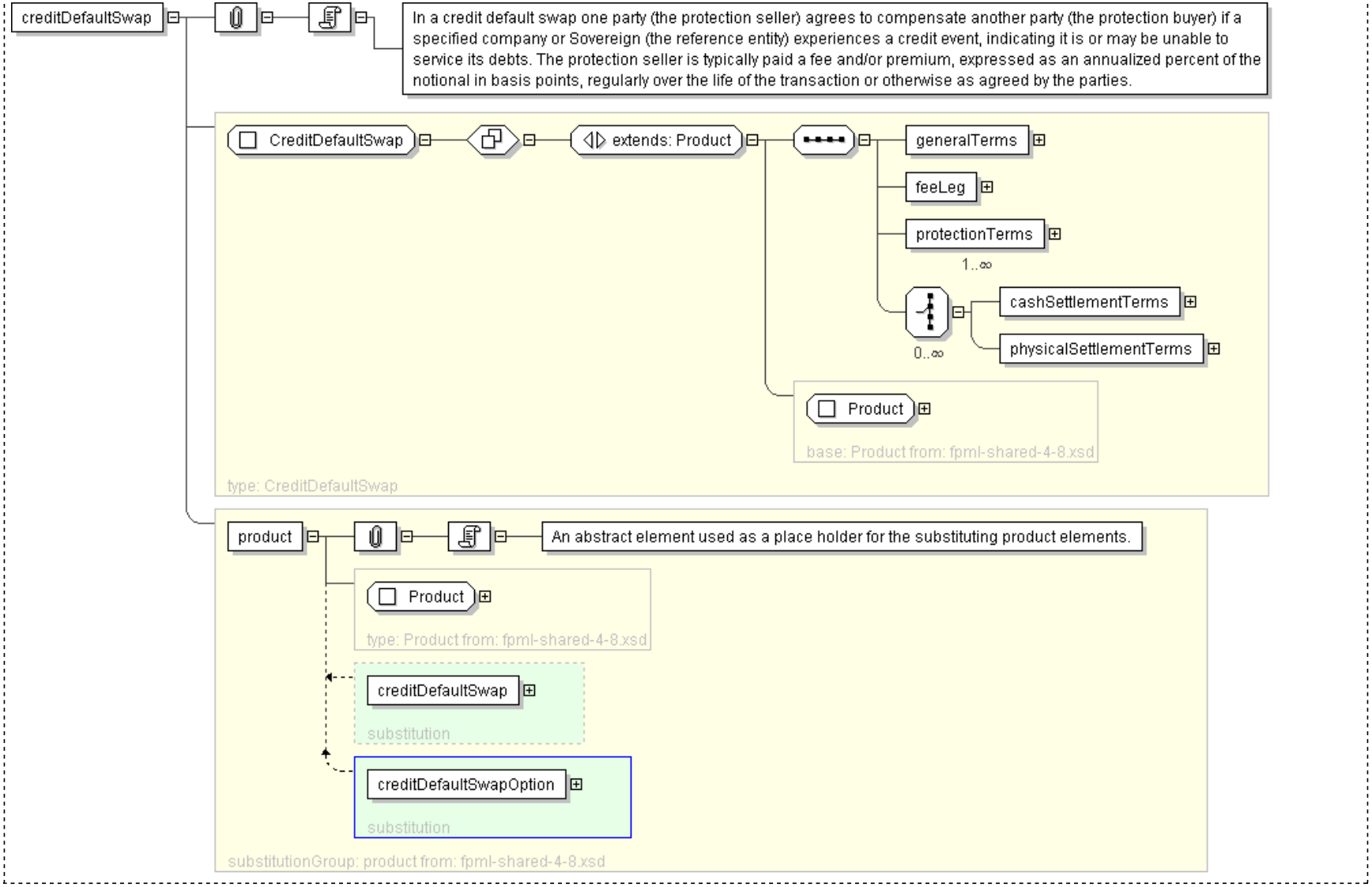

0..1
BusinessCenterTime
valuationTime
http://www.fpml.org/2010/FpML-4-8
0..1
QuotationRateTypeEnum
quotationMethod
http://www.fpml.org/2010/FpML-4-8
0..1
Money
quotationAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Money
minimumQuotationAmount
http://www.fpml.org/2010/FpML-4-8
0..1
string
dealer
http://www.fpml.org/2010/FpML-4-8
0..∞
nonNegativeInteger
cashSettlementBusinessDays
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
Money
cashSettlementAmount
http://www.fpml.org/2010/FpML-4-8
RestrictedPercentage
recoveryFactor
http://www.fpml.org/2010/FpML-4-8
boolean
accruedInterest
http://www.fpml.org/2010/FpML-4-8
0..1
ValuationMethodEnum
valuationMethod
http://www.fpml.org/2010/FpML-4-8
0..1
PhysicalSettlementTerms
physicalSettlementTerms
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency



XML Instance Representation

```
<creditDefaultSwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element.
   Note that the domain values for this element are not strictly an enumerated list.'

   <generalTerms> GeneralTerms </generalTerms> [1]
   'This element contains all the data that appears in the section entitled \"1. General Terms\" in the 2003 ISDA Credit
   Derivatives Confirmation.'

   <feeLeg> FeeLeg </feeLeg> [1]
   'This element contains all the terms relevant to defining the fixed amounts/payments per the applicable ISDA definitions.'

   <protectionTerms> ProtectionTerms </protectionTerms> [1..*]
   'This element contains all the terms relevant to defining the applicable floating rate payer calculation amount, credit
   events and associated conditions to settlement, and reference obligations.'

   Start Choice [0..*]
      <cashSettlementTerms> CashSettlementTerms </cashSettlementTerms> [1]
      'This element contains all the ISDA terms relevant to cash settlement for when cash settlement is applicable. ISDA 2003
      Term: Cash Settlement'

      <physicalSettlementTerms> PhysicalSettlementTerms </physicalSettlementTerms> [1]
      'This element contains all the ISDA terms relevant to physical settlement for when physical settlement is applicable.
      ISDA 2003 Term: Physical Settlement'

   End Choice
</creditDefaultSwap>
```

Diagram



Schema Component Representation

```
<xsd:element name="creditDefaultSwap" type=" CreditDefaultSwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

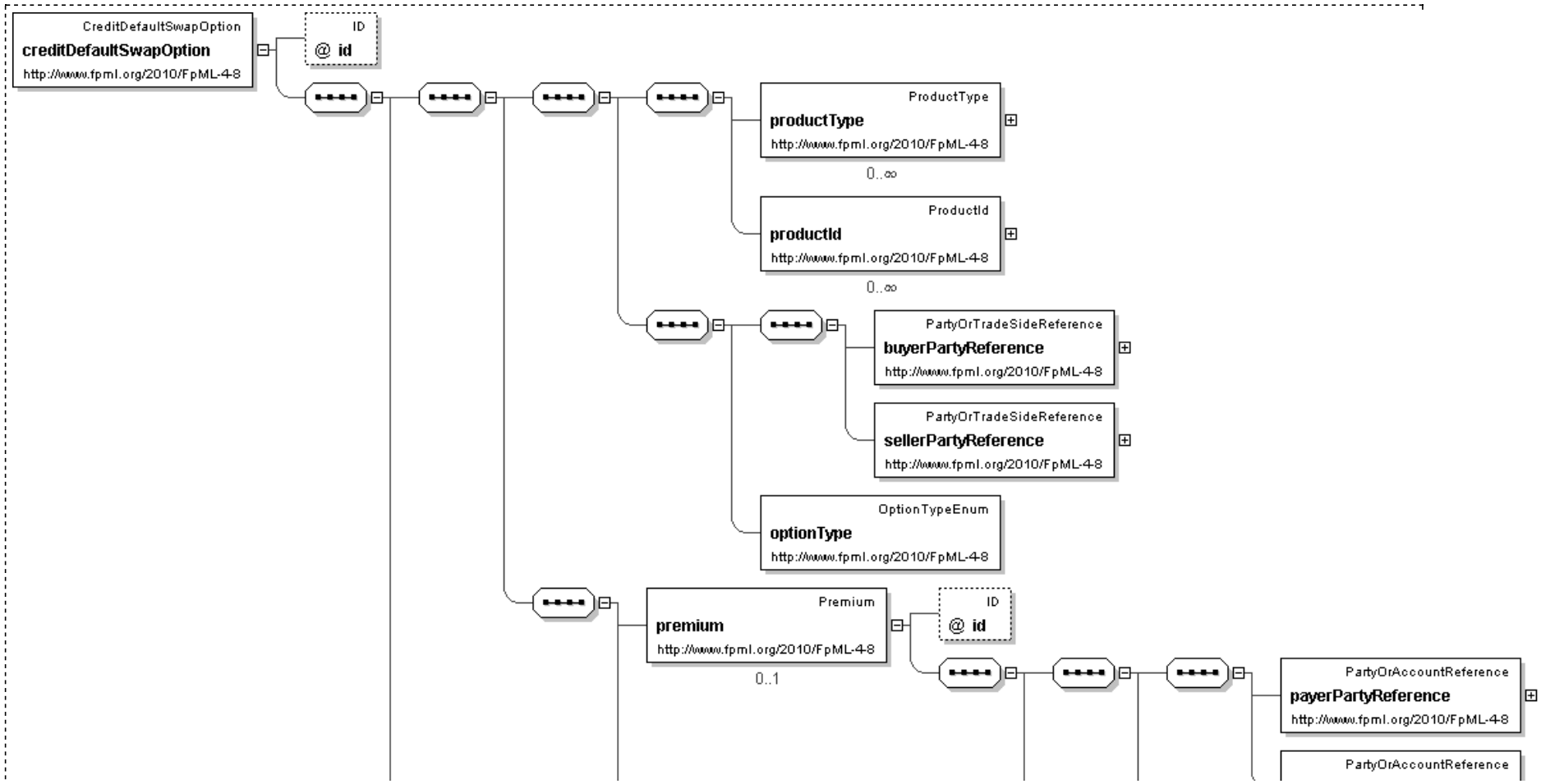
Element: creditDefaultSwapOption

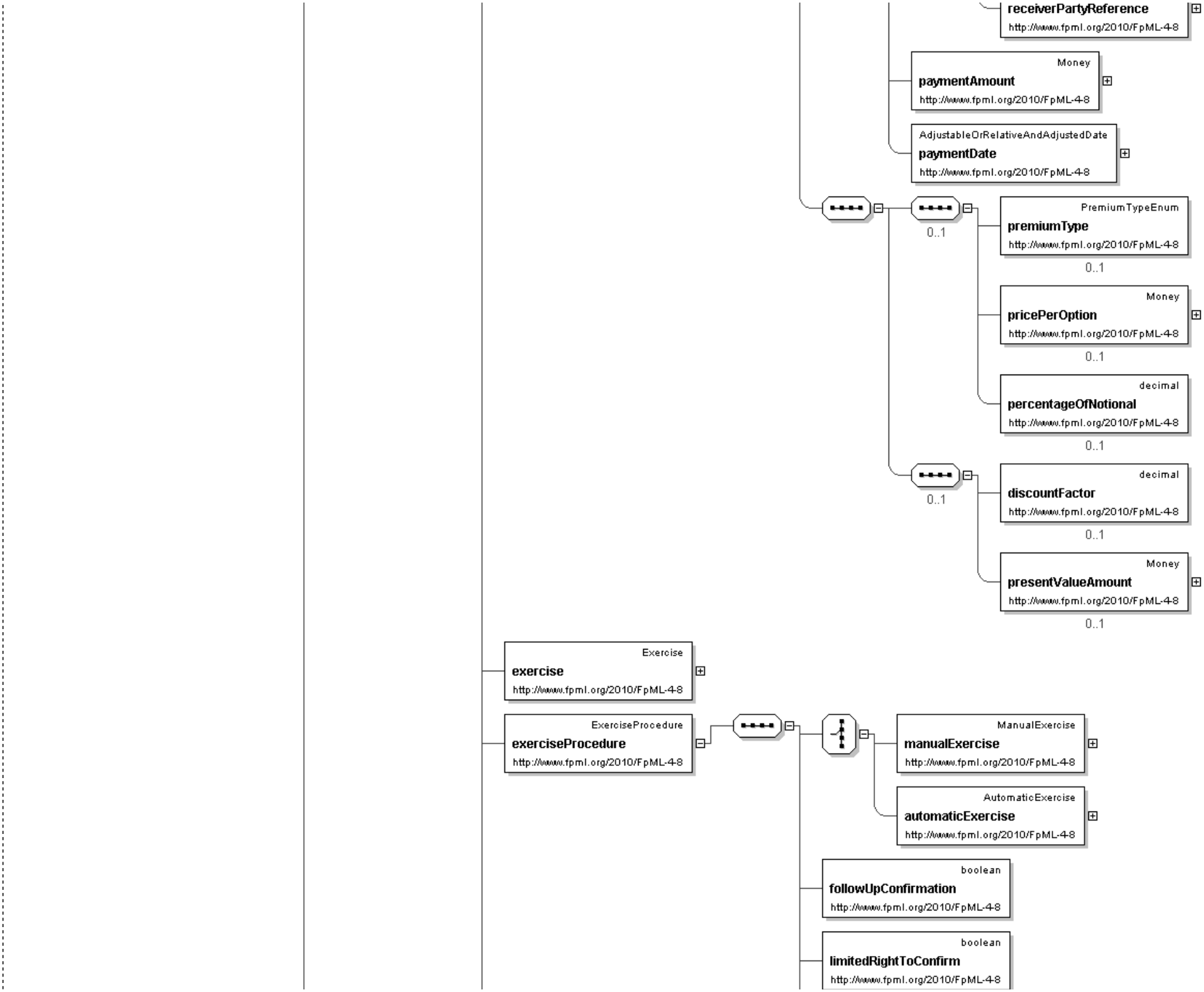
[Table of contents]

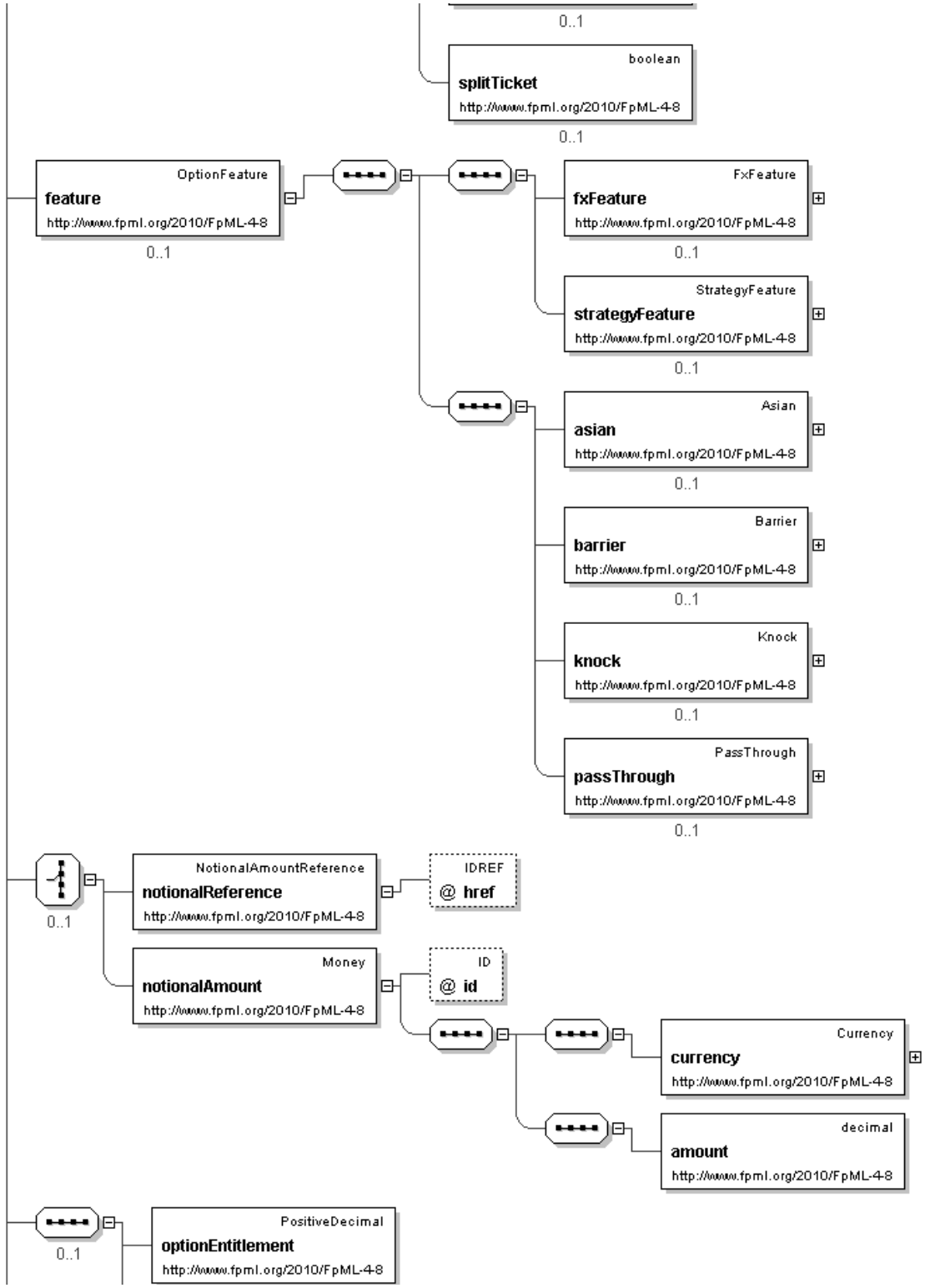
- *This element can be used wherever the following element is referenced:*
  - product

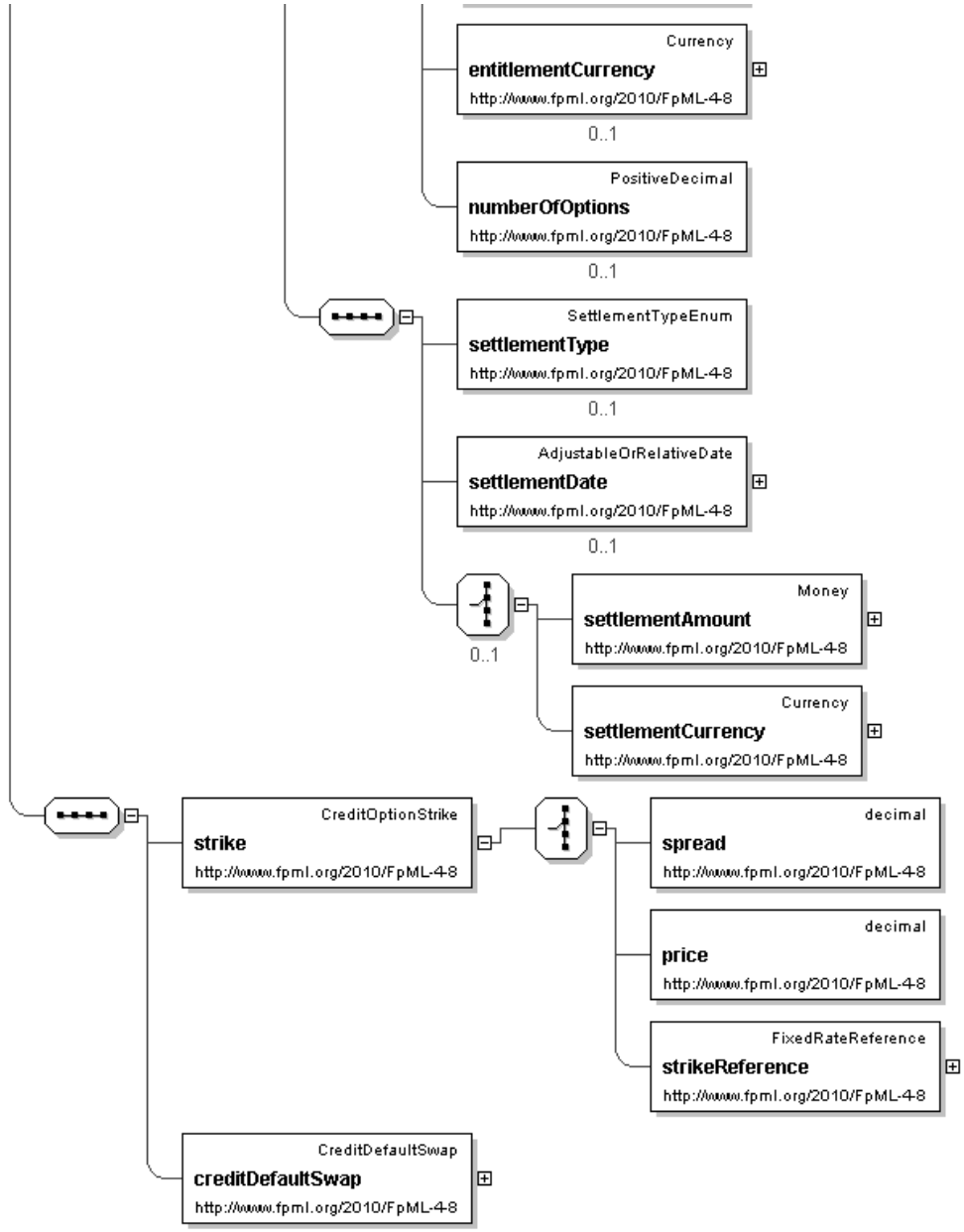
Name	creditDefaultSwapOption
Type	CreditDefaultSwapOption
Nillable	no
Abstract	no
Documentation	An option on a credit default swap.

Logical Diagram




receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeAndAdjustedDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumTypeEnum
premiumType
http://www.fpml.org/2010/FpML-4-8
0..1
Money
pricePerOption
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
percentageOfNotional
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Exercise
exercise
http://www.fpml.org/2010/FpML-4-8
ExerciseProcedure
exerciseProcedure
http://www.fpml.org/2010/FpML-4-8
ManualExercise
manualExercise
http://www.fpml.org/2010/FpML-4-8
AutomaticExercise
automaticExercise
http://www.fpml.org/2010/FpML-4-8
boolean
followUpConfirmation
http://www.fpml.org/2010/FpML-4-8
boolean
limitedRightToConfirm
http://www.fpml.org/2010/FpML-4-8


0..1
boolean
splitTicket
http://www.fpml.org/2010/FpML-4-8
0..1
OptionFeature
feature
http://www.fpml.org/2010/FpML-4-8
0..1
FxFeature
fxFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrategyFeature
strategyFeature
http://www.fpml.org/2010/FpML-4-8
0..1
Asian
asian
http://www.fpml.org/2010/FpML-4-8
0..1
Barrier
barrier
http://www.fpml.org/2010/FpML-4-8
0..1
Knock
knock
http://www.fpml.org/2010/FpML-4-8
0..1
PassThrough
passThrough
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
NotionalAmountReference
notionalReference
http://www.fpml.org/2010/FpML-4-8
IDREF
@ href
Money
notionalAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
0..1
PositiveDecimal
optionEntitlement
http://www.fpml.org/2010/FpML-4-8

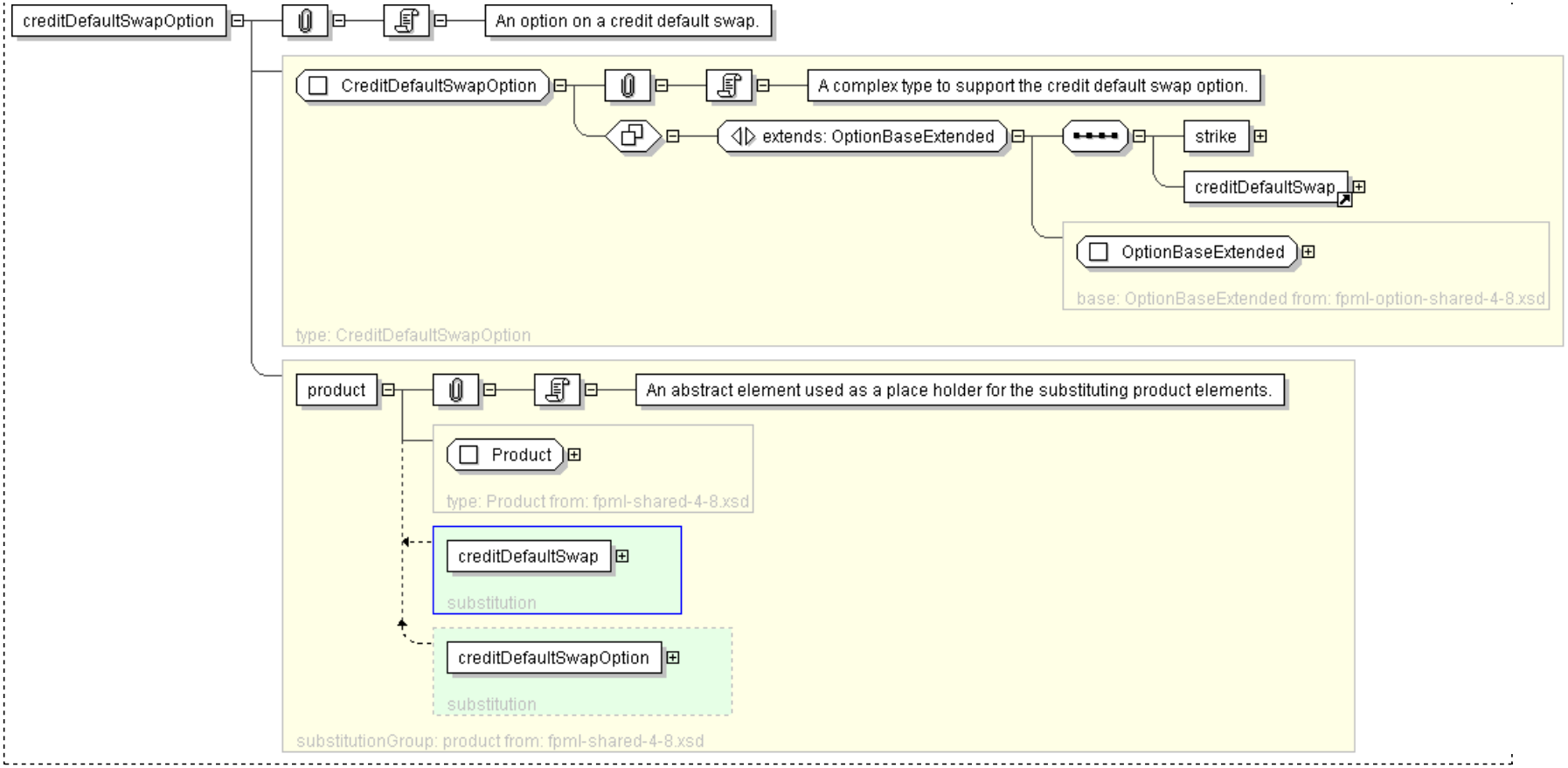


XML Instance Representation

```
<creditDefaultSwapOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
   that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
```

```
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See
   2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and
   in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default option type, while payer/receiver indicator
   is used for options index credit default swaps, consistently with the industry practice. Straddle is used for the case of
   straddle strategy, that combine a call and a put with the same strike.'

   <premium> Premium </premium> [0..1]
   'The option premium payable by the buyer to the seller.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [1]
   'A set of parameters defining procedures associated with the exercise.'

   <feature> OptionFeature </feature> [0..1]
   'An Option feature such as quanto, asian, barrier, knock.'

   Start Choice [0..1]
'A choice between an explicit representation of the notional amount, or a reference to a notional amount defined elsewhere in this
document.'


      <notionalReference> NotionalAmountReference </notionalReference> [1]
      <notionalAmount> Money </notionalAmount> [1]
   End Choice
   Start Group: OptionDenomination.model [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
      'The number of units of underlyer per option comprised in the option transaction.'

      <entitlementCurrency> Currency </entitlementCurrency> [0..1]
      'TODO'

      <numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
      'The number of options comprised in the option transaction.'

   End Group: OptionDenomination.model
   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
         'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
         'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <strike> CreditOptionStrike </strike> [1]
   'Specifies the strike of the option on credit default swap.'

   <creditDefaultSwap> ... </creditDefaultSwap> [1]
</creditDefaultSwapOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="creditDefaultSwapOption" type=" CreditDefaultSwapOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: FixedRecovery.model

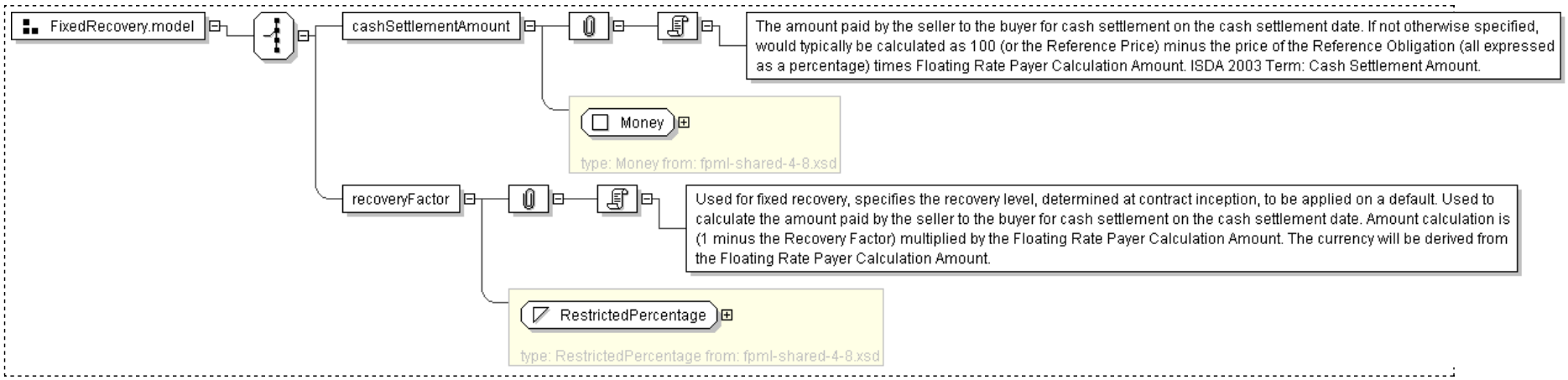
[Table of contents]

Name	FixedRecovery.model
Used by (from the same schema document)	Complex Type CashSettlementTerms

XML Instance Representation

```
Start Choice [1]
   <cashSettlementAmount> Money </cashSettlementAmount> [1]
   'The amount paid by the seller to the buyer for cash settlement on the cash settlement date. If not otherwise specified, would typically be
   calculated as 100 (or the Reference Price) minus the price of the Reference Obligation (all expressed as a percentage) times Floating Rate
   Payer Calculation Amount. ISDA 2003 Term: Cash Settlement Amount.'

   <recoveryFactor> RestrictedPercentage </recoveryFactor> [1]
   'Used for fixed recovery, specifies the recovery level, determined at contract inception, to be applied on a default. Used to calculate the
   amount paid by the seller to the buyer for cash settlement on the cash settlement date. Amount calculation is (1 minus the Recovery Factor)
   multiplied by the Floating Rate Payer Calculation Amount. The currency will be derived from the Floating Rate Payer Calculation Amount.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="FixedRecovery.model">
  <xsd:choice>
    <xsd:element name="cashSettlementAmount" type=" Money "/>
    <xsd:element name="recoveryFactor" type=" RestrictedPercentage "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdditionalFixedPayments

[Table of contents]

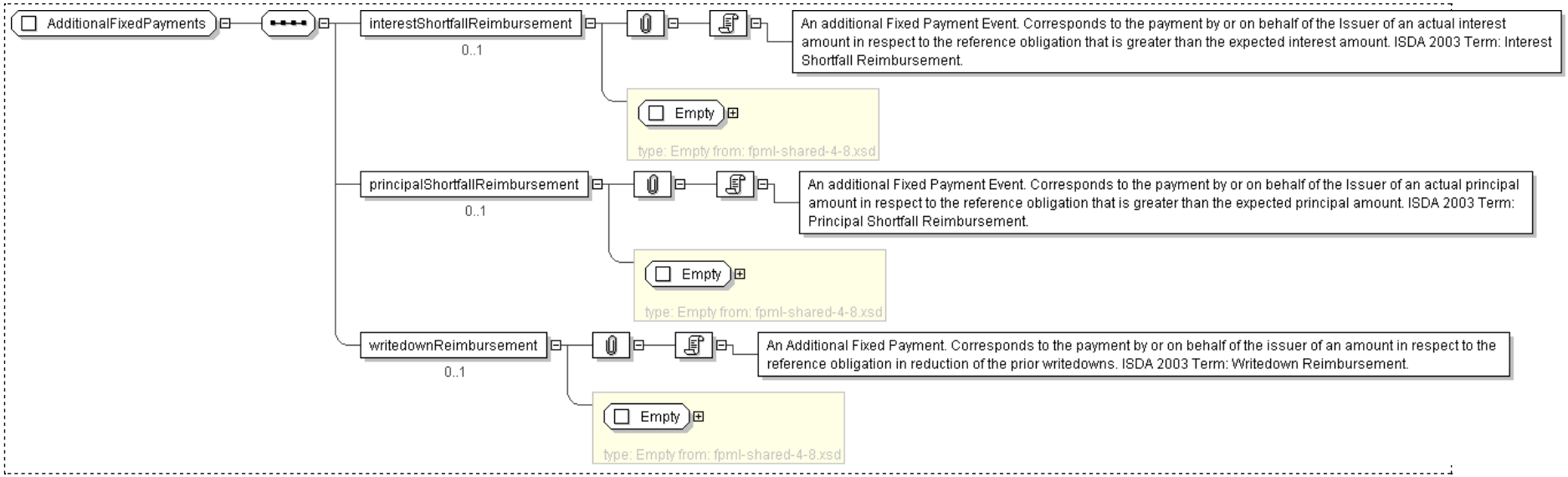
Super-types:	None
Sub-types:	None

Name	AdditionalFixedPayments
Used by (from the same schema document)	Complex Type FloatingAmountEvents
Abstract	no

XML Instance Representation

```
<...>
   <interestShortfallReimbursement> Empty </interestShortfallReimbursement> [0..1]
   'An additional Fixed Payment Event. Corresponds to the payment by or on behalf of the Issuer of an actual interest amount in respect to the reference
   obligation that is greater than the expected interest amount. ISDA 2003 Term: Interest Shortfall Reimbursement.'

   <principalShortfallReimbursement> Empty </principalShortfallReimbursement> [0..1]
   'An additional Fixed Payment Event. Corresponds to the payment by or on behalf of the Issuer of an actual principal amount in respect to the reference
   obligation that is greater than the expected principal amount. ISDA 2003 Term: Principal Shortfall Reimbursement.'

   <writedownReimbursement> Empty </writedownReimbursement> [0..1]
   'An Additional Fixed Payment. Corresponds to the payment by or on behalf of the issuer of an amount in respect to the reference obligation in reduction
   of the prior writedowns. ISDA 2003 Term: Writedown Reimbursement.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdditionalFixedPayments">
   <xsd:sequence>
      <xsd:element name="interestShortfallReimbursement" type=" Empty " minOccurs="0"/>
      <xsd:element name="principalShortfallReimbursement" type=" Empty " minOccurs="0"/>
      <xsd:element name="writedownReimbursement" type=" Empty " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdditionalTerm

[Table of contents]

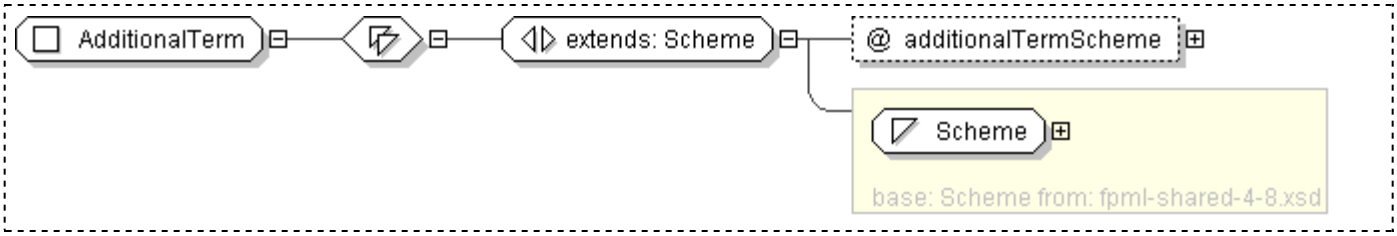
Super-types:	Scheme < AdditionalTerm (by extension)
Sub-types:	None

Name	AdditionalTerm
Used by (from the same schema document)	Complex Type GeneralTerms
Abstract	no

XML Instance Representation

```
<...
 additionalTermScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdditionalTerm">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="additionalTermScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustedPaymentDates

[Table of contents]

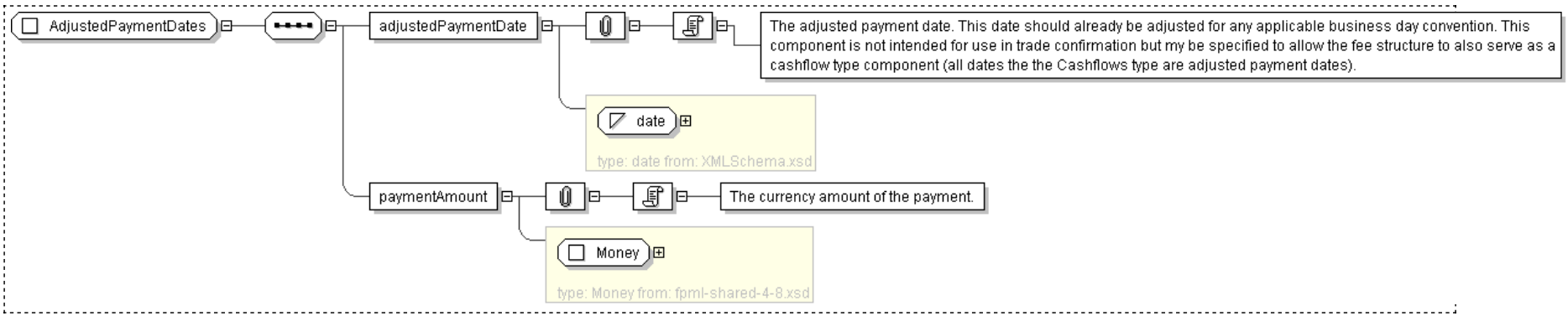
Super-types:	None
Sub-types:	None

Name	AdjustedPaymentDates
Used by (from the same schema document)	Complex Type PeriodicPayment
Abstract	no

XML Instance Representation

```
<...>
   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
   'The adjusted payment date. This date should already be adjusted for any applicable business day convention. This component is not intended
   for use in trade confirmation but my be specified to allow the fee structure to also serve as a cashflow type component (all dates the the
   Cashflows type are adjusted payment dates).'

   <paymentAmount> Money </paymentAmount> [1]
   'The currency amount of the payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustedPaymentDates">
   <xsd:sequence>
      <xsd:element name="adjustedPaymentDate" type=" xsd:date "/>
      <xsd:element name="paymentAmount" type=" Money "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasketReferenceInformation

[Table of contents]

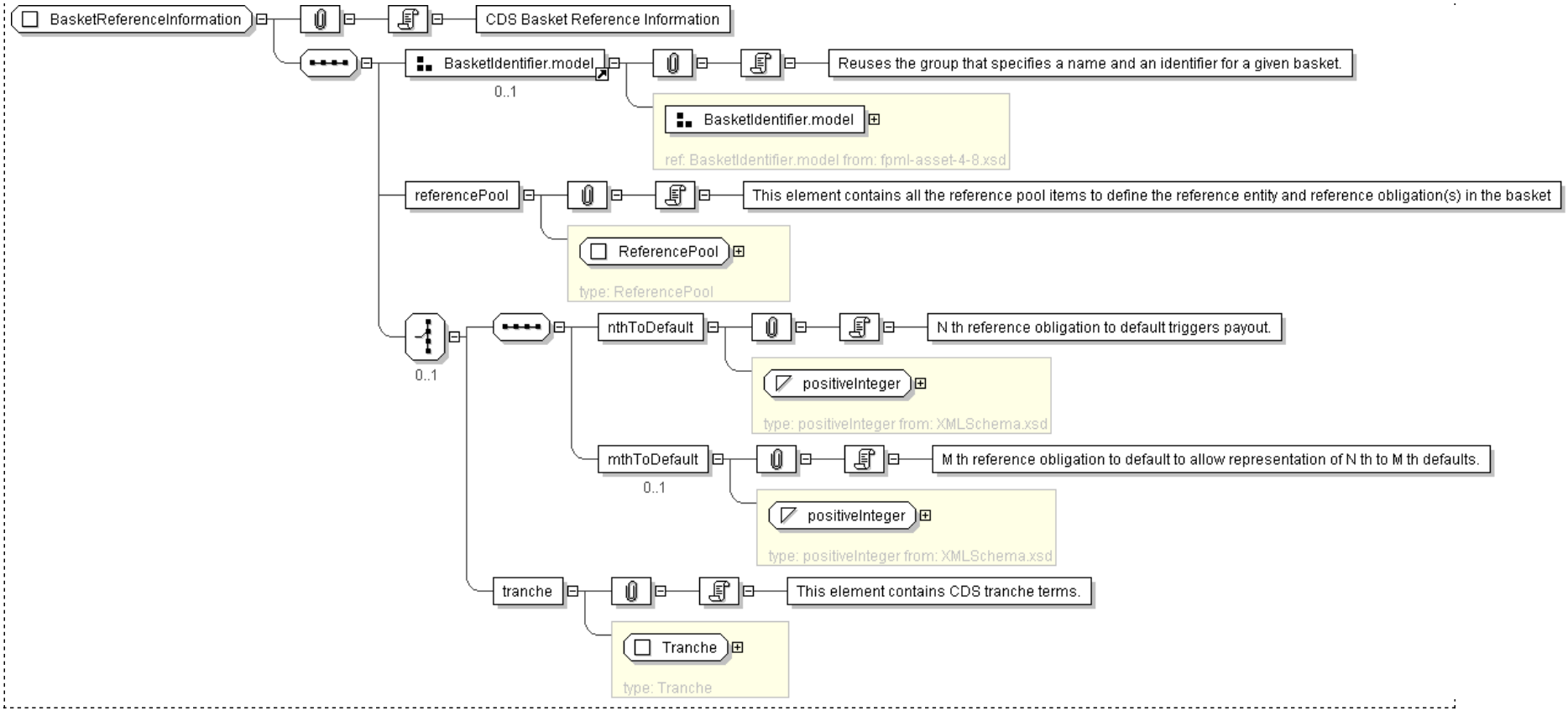
Super-types:	None
Sub-types:	None

Name	BasketReferenceInformation
Used by (from the same schema document)	Complex Type GeneralTerms
Abstract	no
Documentation	CDS Basket Reference Information

XML Instance Representation

```
<...>
  Start Group: BasketIdentifier.model [0..1]
'Reuses the group that specifies a name and an identifier for a given basket.'


    Start Choice [1]
      <basketName> BasketName </basketName> [1]
      'The name of the basket expressed as a free format string. FpML does not define usage rules for this element.'

      <basketId> BasketId </basketId> [0..*]
      'A CDS basket identifier'

      <basketId> BasketId </basketId> [1..*]
      'A CDS basket identifier'

    End Choice
  End Group: BasketIdentifier.model
  <referencePool> ReferencePool </referencePool> [1]
  'This element contains all the reference pool items to define the reference entity and reference obligation(s) in the basket'

  Start Choice [0..1]
    <nthToDefault> xsd:positiveInteger </nthToDefault> [1]
    'N th reference obligation to default triggers payout.'

    <mthToDefault> xsd:positiveInteger </mthToDefault> [0..1]
    'M th reference obligation to default to allow representation of N th to M th defaults.'

    <tranche> Tranche </tranche> [1]
    'This element contains CDS tranche terms.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasketReferenceInformation">
  <xsd:sequence>
    <xsd:group ref=" BasketIdentifier.model " minOccurs="0"/>
    <xsd:element name="referencePool" type=" ReferencePool "/>
    <xsd:choice minOccurs="0">
      <xsd:sequence>
        <xsd:element name="nthToDefault" type=" xsd:positiveInteger "/>
        <xsd:element name="mthToDefault" type=" xsd:positiveInteger " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="tranche" type=" Tranche "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationAmount

[Table of contents]

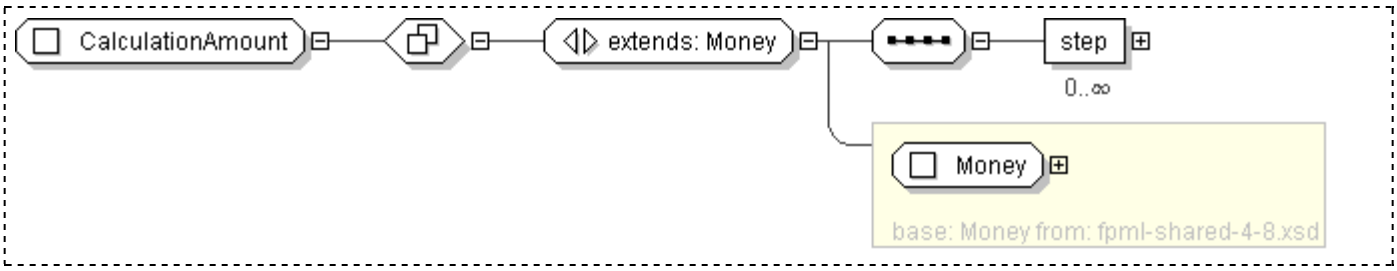
Super-types:	Money < CalculationAmount (by extension)
Sub-types:	None

Name	CalculationAmount
Used by (from the same schema document)	Complex Type FixedAmountCalculation
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

   <amount> xsd:decimal </amount> [1]
   'The monetary quantity in currency units.'

   <step> Step </step> [0..*]
   'A schedule of step date and value pairs. On each step date the associated step
   value becomes effective. A list of steps may be ordered in the document by
   ascending step date. An FpML document containing an unordered list of steps is
   still regarded as a conformant document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationAmount">
  <xsd:complexContent>
    <xsd:extension base=" Money ">
      <xsd:sequence>
        <xsd:element name="step" type=" Step " minOccurs="0"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashSettlementTerms

[Table of contents]

Super-types:	SettlementTerms < CashSettlementTerms (by extension)
Sub-types:	None

Name	CashSettlementTerms
Used by (from the same schema document)	Complex Type CreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <settlementCurrency> Currency </settlementCurrency> [0..1]
   'ISDA 2003 Term: Settlement Currency'

   <valuationDate> ValuationDate </valuationDate> [0..1]
   'The number of business days after conditions to settlement have been satisfied when
   the calculation agent obtains a price quotation on the Reference Obligation for
   purposes of cash settlement. There may be one or more valuation dates. This is
   typically specified if the cash settlement amount is not a fixed amount. ISDA 2003
   Term: Valuation Date'

   <valuationTime> BusinessCenterTime </valuationTime> [0..1]
   'The time of day in the specified business center when the calculation agent seeks
   quotations for an amount of the reference obligation for purposes of cash settlement.
   ISDA 2003 Term: Valuation Time'

   <quotationMethod> QuotationRateTypeEnum </quotationMethod> [0..1]
   'The type of price quotations to be requested from dealers when determining the market
   value of the reference obligation for purposes of cash settlement. For example, Bid,
   Offer or Mid-market. ISDA 2003 Term: Quotation Method'

   <quotationAmount> Money </quotationAmount> [0..1]
   'In the determination of a cash settlement amount, if weighted average quotations are
   to be obtained, the quotation amount specifies an upper limit to the outstanding
   principal balance of the reference obligation for which the quote should be obtained.
   If not specified, the ISDA definitions provide for a fallback amount equal to the
   floating rate payer calculation amount. ISDA 2003 Term: Quotation Amount'

   <minimumQuotationAmount> Money </minimumQuotationAmount> [0..1]
   'In the determination of a cash settlement amount, if weighted average quotations are
   to be obtained, the minimum quotation amount specifies a minimum intended threshold
   amount of outstanding principal balance of the reference obligation for which the
   quote should be obtained. If not specified, the ISDA definitions provide for a
   fallback amount of the lower of either USD 1,000,000 (or its equivalent in the
   relevant obligation currency) or the quotation amount. ISDA 2003 Term: Minimum
   Quotation Amount'

   <dealer> xsd:string </dealer> [0..*]
   'A dealer from whom quotations are obtained by the calculation agent on the reference
   obligation for purposes of cash settlement. ISDA 2003 Term: Dealer'

   <cashSettlementBusinessDays> xsd:nonNegativeInteger </cashSettlementBusinessDays> [0..1]
   'The number of business days used in the determination of the cash settlement payment
   date. If a cash settlement amount is specified, the cash settlement payment date will
   be this number of business days following the calculation of the final price. If a
   cash settlement amount is not specified, the cash settlement payment date will be this
   number of business days after all conditions to settlement are satisfied. ISDA 2003
   Term: Cash Settlement Date'

   Start Group: FixedRecovery.model [0..1]
     Start Choice [1]
```

<cashSettlementAmount> Money **</cashSettlementAmount>** [1]

'The amount paid by the seller to the buyer for cash settlement on the cash settlement date. If not otherwise specified, would typically be calculated as 100 (or the Reference Price) minus the price of the Reference Obligation (all expressed as a percentage) times Floating Rate Payer Calculation Amount. ISDA 2003 Term: Cash Settlement Amount.'

<recoveryFactor> RestrictedPercentage **</recoveryFactor>** [1]

'Used for fixed recovery, specifies the recovery level, determined at contract inception, to be applied on a default. Used to calculate the amount paid by the seller to the buyer for cash settlement on the cash settlement date. Amount calculation is (1 minus the Recovery Factor) multiplied by the Floating Rate Payer Calculation Amount. The currency will be derived from the Floating Rate Payer Calculation Amount.'

End Choice

End Group: FixedRecovery.model

<accruedInterest> xsd:boolean **</accruedInterest>** [0..1]

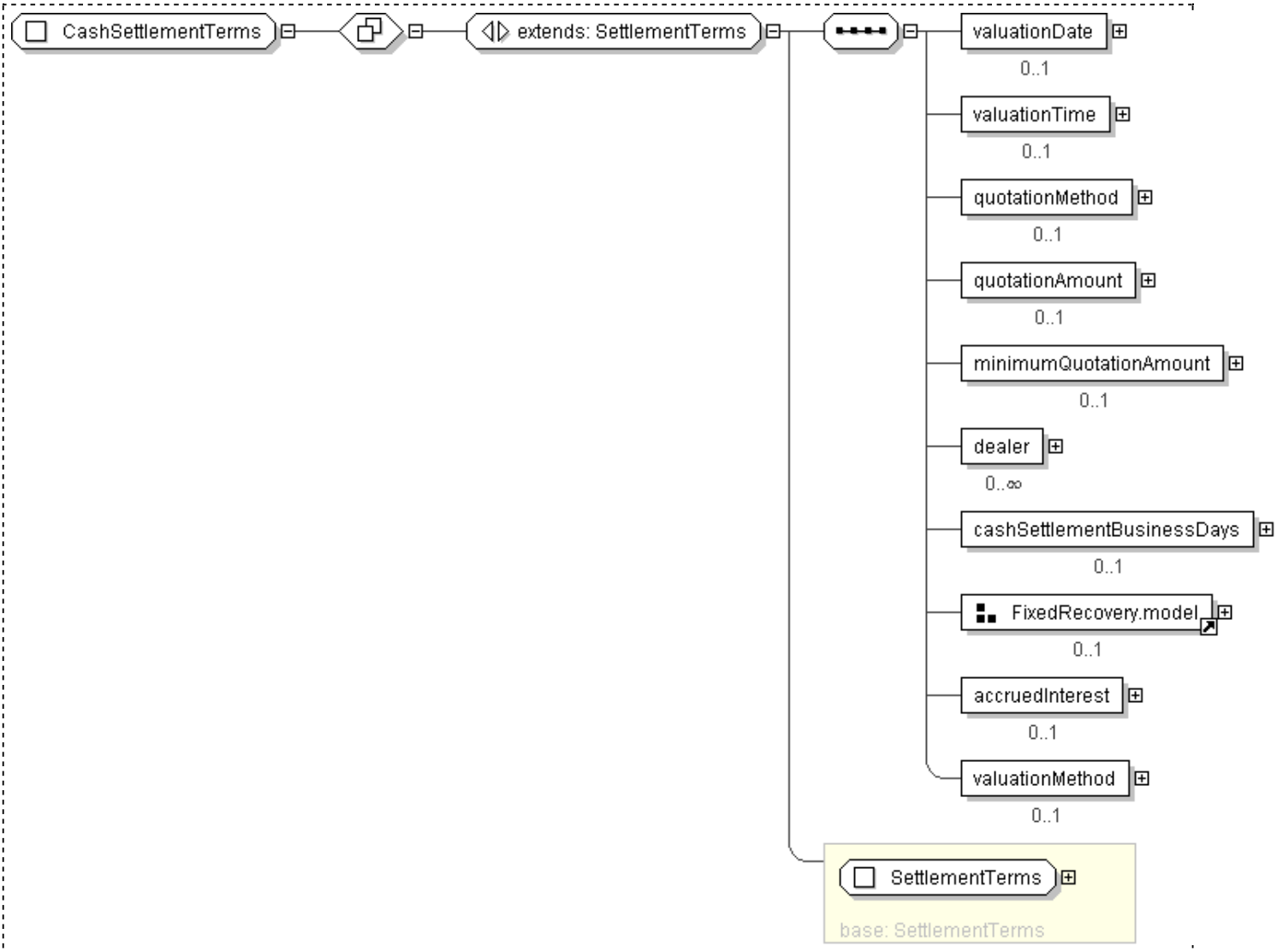
'Indicates whether accrued interest is included (true) or not (false). For cash settlement this specifies whether quotations should be obtained inclusive or not of accrued interest. For physical settlement this specifies whether the buyer should deliver the obligation with an outstanding principal balance that includes or excludes accrued interest. ISDA 2003 Term: Include/Exclude Accrued Interest'

<valuationMethod> ValuationMethodEnum **</valuationMethod>** [0..1]

'The ISDA defined methodology for determining the final price of the reference obligation for purposes of cash settlement. (ISDA 2003 Term: Valuation Method). For example, Market, Highest etc.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="CashSettlementTerms">
  <xsd:complexContent>
    <xsd:extension base=" SettlementTerms ">
      <xsd:sequence>
        <xsd:element name="valuationDate" type=" ValuationDate " minOccurs="0"/>
        <xsd:element name="valuationTime" type=" BusinessCenterTime " minOccurs="0"/>
        <xsd:element name="quotationMethod" type=" QuotationRateTypeEnum " minOccurs="0"/>
        <xsd:element name="quotationAmount" type=" Money " minOccurs="0"/>
        <xsd:element name="minimumQuotationAmount" type=" Money " minOccurs="0"/>
        <xsd:element name="dealer" type=" xsd:string " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="cashSettlementBusinessDays" type=" xsd:nonNegativeInteger "
        minOccurs="0"/>
        <xsd:group ref=" FixedRecovery.model " minOccurs="0"/>
        <xsd:element name="accruedInterest" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="valuationMethod" type=" ValuationMethodEnum " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditDefaultSwap

[Table of contents]

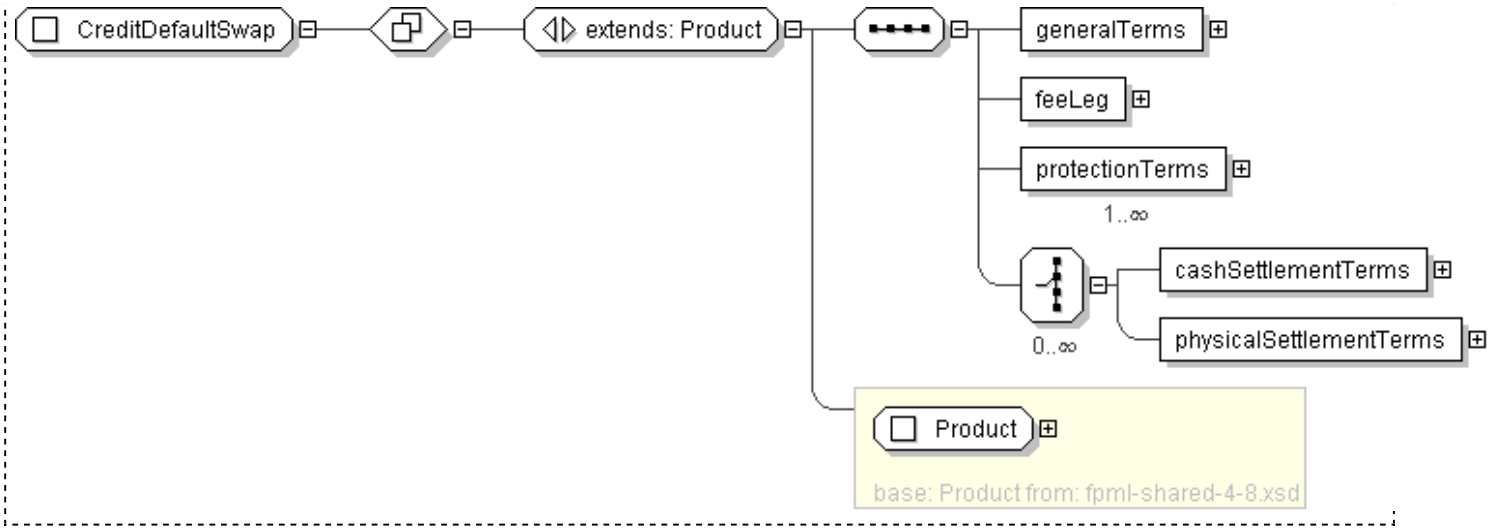
Super-types:	Product < CreditDefaultSwap (by extension)
Sub-types:	None

Name	CreditDefaultSwap
Used by (from the same schema document)	Element creditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for this
   element are not strictly an enumerated list.'

   <generalTerms> GeneralTerms </generalTerms> [1]
   'This element contains all the data that appears in the section entitled \"1.
   General Terms\" in the 2003 ISDA Credit Derivatives Confirmation.'

   <feeLeg> FeeLeg </feeLeg> [1]
   'This element contains all the terms relevant to defining the fixed
   amounts/payments per the applicable ISDA definitions.'

   <protectionTerms> ProtectionTerms </protectionTerms> [1..*]
   'This element contains all the terms relevant to defining the applicable floating
   rate payer calculation amount, credit events and associated conditions to
   settlement, and reference obligations.'

   Start Choice [0..*]
      <cashSettlementTerms> CashSettlementTerms </cashSettlementTerms> [1]
      'This element contains all the ISDA terms relevant to cash settlement for when
      cash settlement is applicable. ISDA 2003 Term: Cash Settlement'

      <physicalSettlementTerms> PhysicalSettlementTerms </physicalSettlementTerms> [1]
      'This element contains all the ISDA terms relevant to physical settlement for
      when physical settlement is applicable. ISDA 2003 Term: Physical Settlement'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditDefaultSwap">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="generalTerms" type=" GeneralTerms "/>
            <xsd:element name="feeLeg" type=" FeeLeg "/>
            <xsd:element name="protectionTerms" type=" ProtectionTerms "
            maxOccurs="unbounded"/>
            <xsd:choice minOccurs="0" maxOccurs="unbounded">
               <xsd:element name="cashSettlementTerms" type=" CashSettlementTerms "/>
               <xsd:element name="physicalSettlementTerms" type=" PhysicalSettlementTerms
               "/>
            </xsd:choice>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditDefaultSwapOption

[Table of contents]

Super-types:	OptionBaseExtended < CreditDefaultSwapOption (by extension)
Sub-types:	None

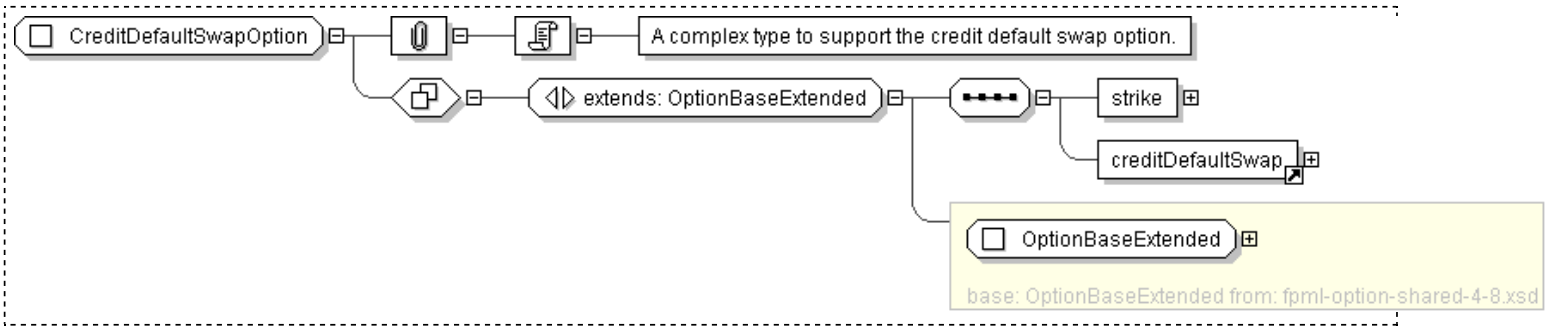
Name	CreditDefaultSwapOption
Used by (from the same schema document)	Element creditDefaultSwapOption
Abstract	no
Documentation	A complex type to support the credit default swap option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a
   coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not strictly an
   enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives
   the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this
   the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights
   defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions
   Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default option type,
   while payer/receiver indicator is used for options index credit default swaps, consistently
   with the industry practice. Straddle is used for the case of straddle strategy, that combine a
   call and a put with the same strike.'

   <premium> Premium </premium> [0..1]
   'The option premium payable by the buyer to the seller.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [1]
   'A set of parameters defining procedures associated with the exercise.'

   <feature> OptionFeature </feature> [0..1]
   'An Option feature such as quanto, asian, barrier, knock.'

   Start Choice [0..1]
'A choice between an explicit representation of the notional amount, or a reference to a notional
amount defined elsewhere in this document.'


      <notionalReference> NotionalAmountReference </notionalReference> [1]
      <notionalAmount> Money </notionalAmount> [1]
   End Choice
   Start Group: OptionDenomination.model [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
      'The number of units of underlyer per option comprised in the option transaction.'

      <entitlementCurrency> Currency </entitlementCurrency> [0..1]
      'TODO'

      <numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
      'The number of options comprised in the option transaction.'

   End Group: OptionDenomination.model
   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
```

```
      Start Choice [1]
        <settlementAmount> Money </settlementAmount> [1]
          'Settlement Amount'

        <settlementCurrency> Currency </settlementCurrency> [1]
          'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
    End Group: SettlementAmountOrCurrency.model
    <strike> CreditOptionStrike </strike> [1]
    'Specifies the strike of the option on credit default swap.'

    <creditDefaultSwap> ... </creditDefaultSwap> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditDefaultSwapOption">
  <xsd:complexContent>
    <xsd:extension base=" OptionBaseExtended ">
      <xsd:sequence>
        <xsd:element name="strike" type=" CreditOptionStrike "/>
        <xsd:element ref=" creditDefaultSwap "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditOptionStrike

[Table of contents]

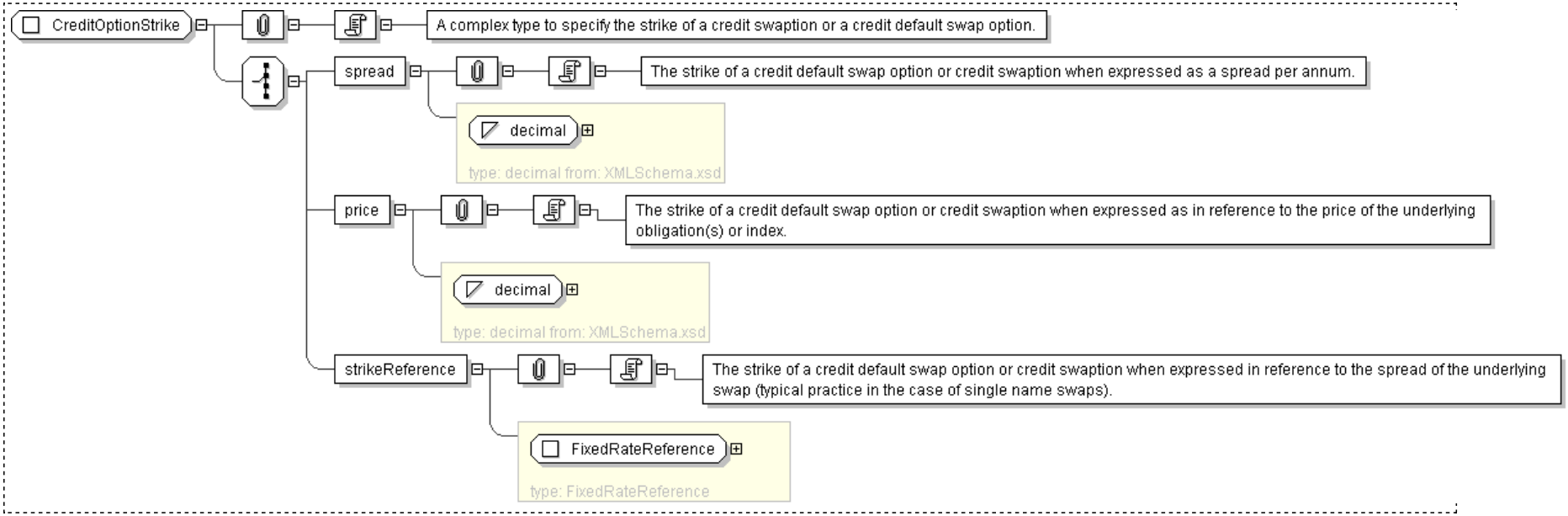
Super-types:	None
Sub-types:	None

Name	CreditOptionStrike
Used by (from the same schema document)	Complex Type CreditDefaultSwapOption
Abstract	no
Documentation	A complex type to specify the strike of a credit swaption or a credit default swap option.

XML Instance Representation

```
<...>
  Start Choice [1]
    <spread> xsd:decimal </spread> [1]
      'The strike of a credit default swap option or credit swaption when expressed as a spread per annum.'

    <price> xsd:decimal </price> [1]
      'The strike of a credit default swap option or credit swaption when expressed as in reference to the price of the underlying
      obligation(s) or index.'

    <strikeReference> FixedRateReference </strikeReference> [1]
      'The strike of a credit default swap option or credit swaption when expressed in reference to the spread of the underlying swap
      (typical practice in the case of single name swaps).'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditOptionStrike">
  <xsd:choice>
    <xsd:element name="spread" type=" xsd:decimal "/>
    <xsd:element name="price" type=" xsd:decimal "/>
    <xsd:element name="strikeReference" type=" FixedRateReference "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeliverableObligations

[Table of contents]

Super-types:	None
Sub-types:	None

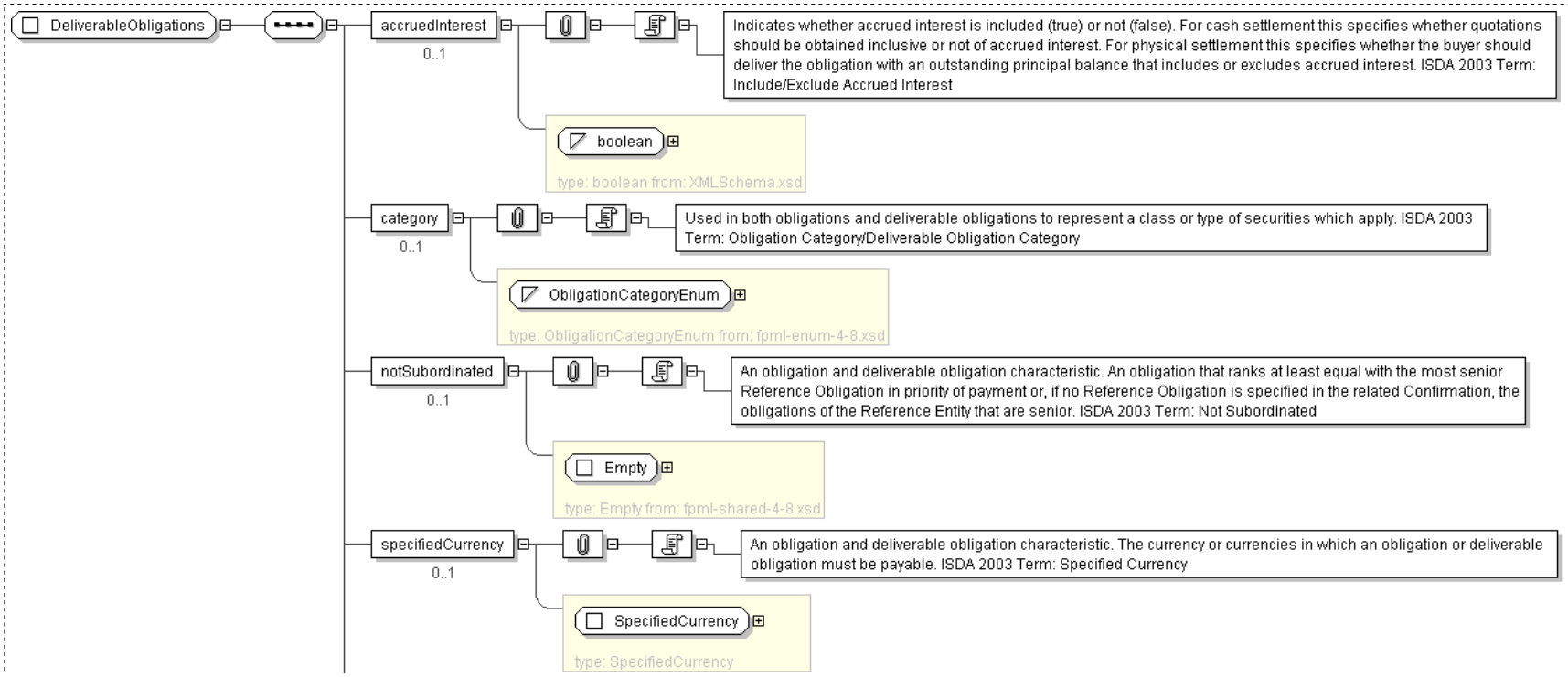
Name	DeliverableObligations
Used by (from the same schema document)	Complex Type PhysicalSettlementTerms
Abstract	no

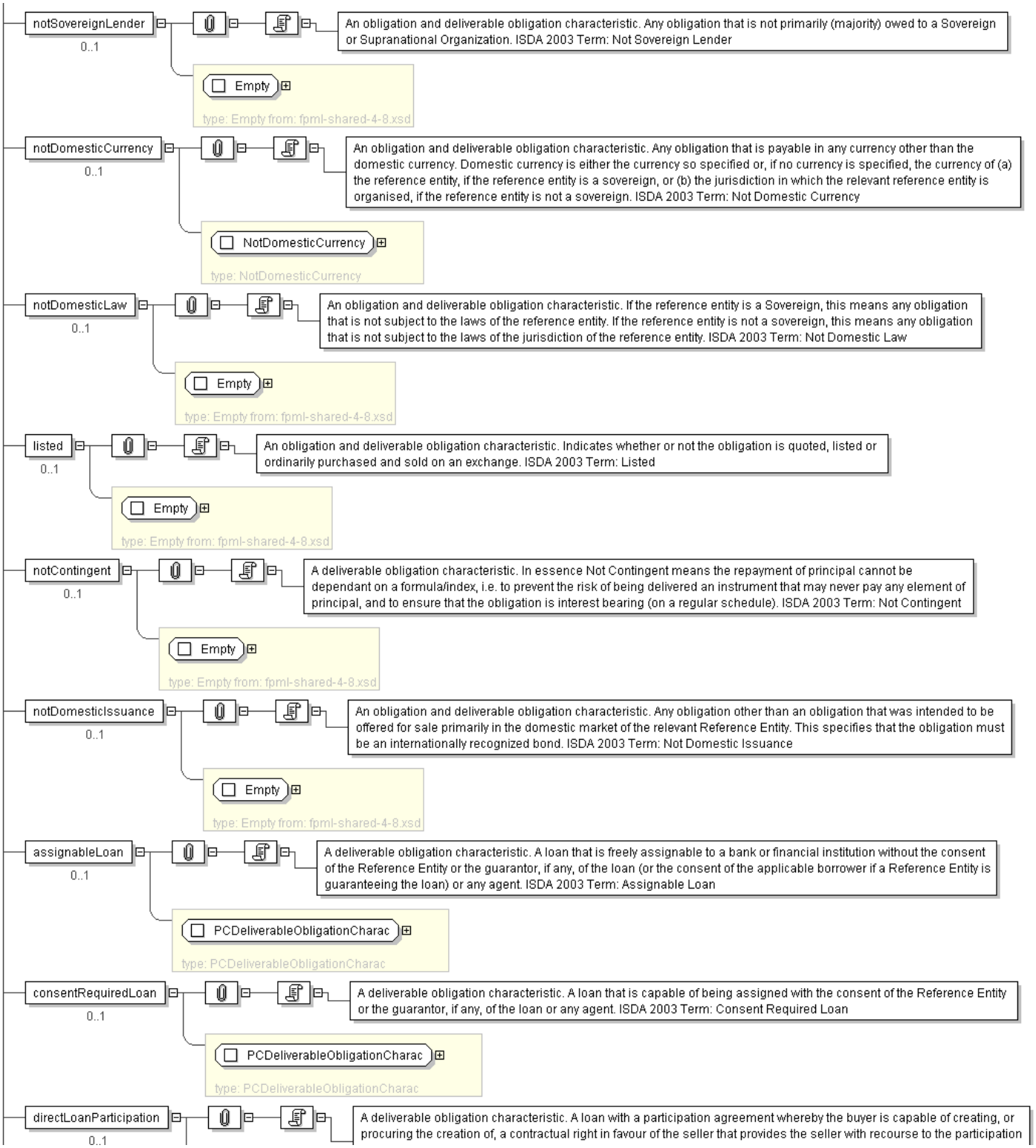
XML Instance Representation

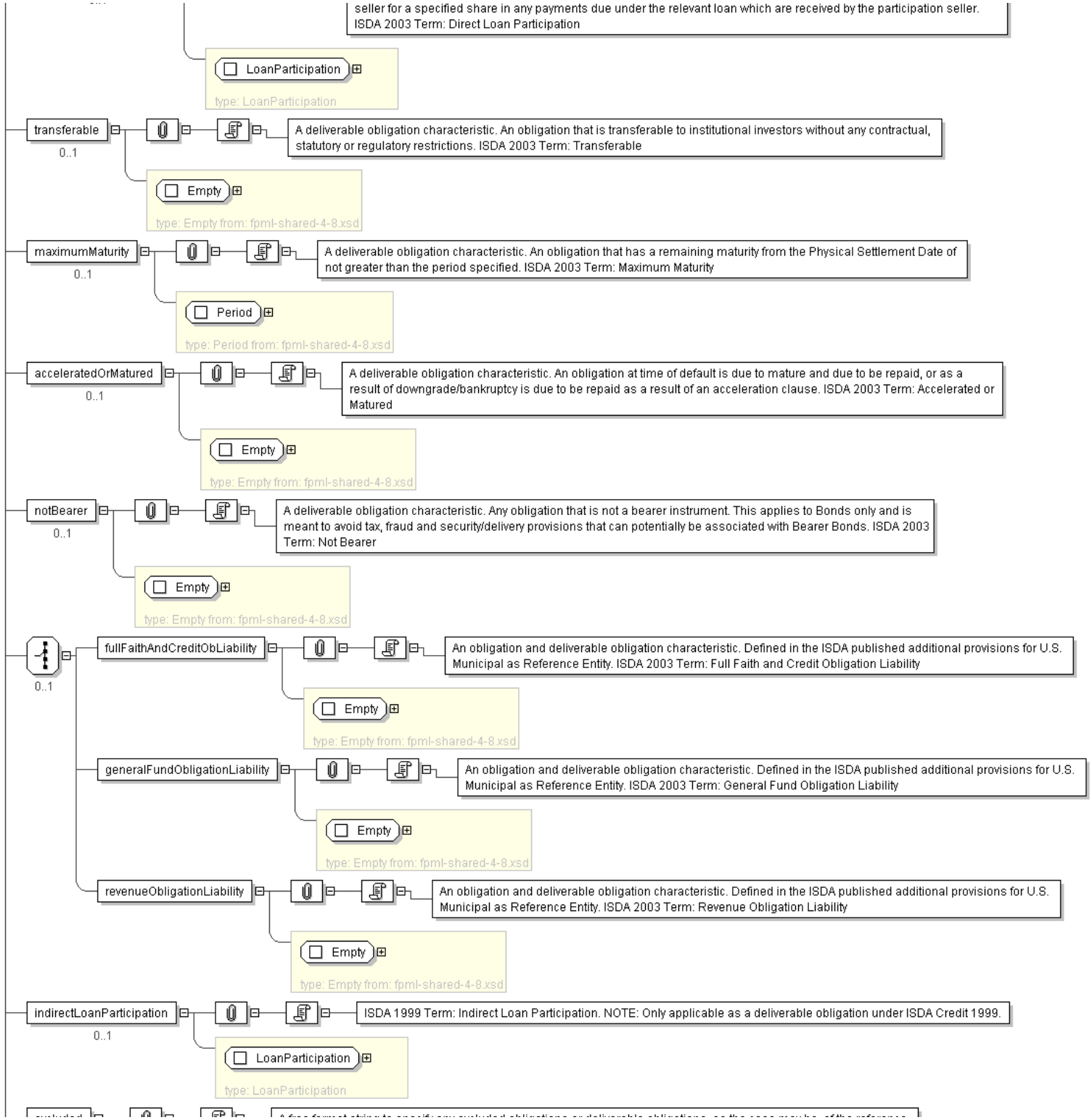
```
<...>
   <accruedInterest> xsd:boolean </accruedInterest> [0..1]
   'Indicates whether accrued interest is included (true) or not (false). For cash settlement this specifies whether quotations should be obtained inclusive
   or not of accrued interest. For physical settlement this specifies whether the buyer should deliver the obligation with an outstanding principal balance
   that includes or excludes accrued interest. ISDA 2003 Term: Include/Exclude Accrued Interest'

   <category> ObligationCategoryEnum </category> [0..1]
   'Used in both obligations and deliverable obligations to represent a class or type of securities which apply. ISDA 2003 Term: Obligation
   Category/Deliverable Obligation Category'

   <notSubordinated> Empty </notSubordinated> [0..1]
   'An obligation and deliverable obligation characteristic. An obligation that ranks at least equal with the most senior Reference Obligation in priority
   of payment or, if no Reference Obligation is specified in the related Confirmation, the obligations of the Reference Entity that are senior. ISDA 2003
   Term: Not Subordinated'

   <specifiedCurrency> SpecifiedCurrency </specifiedCurrency> [0..1]
   'An obligation and deliverable obligation characteristic. The currency or currencies in which an obligation or deliverable obligation must be payable.
   ISDA 2003 Term: Specified Currency'

   <notSovereignLender> Empty </notSovereignLender> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation that is not primarily (majority) owed to a Sovereign or Supranational
   Organization. ISDA 2003 Term: Not Sovereign Lender'

   <notDomesticCurrency> NotDomesticCurrency </notDomesticCurrency> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation that is payable in any currency other than the domestic currency. Domestic
   currency is either the currency so specified or, if no currency is specified, the currency of (a) the reference entity, if the reference entity is a
   sovereign, or (b) the jurisdiction in which the relevant reference entity is organised, if the reference entity is not a sovereign. ISDA 2003 Term: Not
   Domestic Currency'

   <notDomesticLaw> Empty </notDomesticLaw> [0..1]
   'An obligation and deliverable obligation characteristic. If the reference entity is a Sovereign, this means any obligation that is not subject to the
   laws of the reference entity. If the reference entity is not a sovereign, this means any obligation that is not subject to the laws of the jurisdiction
   of the reference entity. ISDA 2003 Term: Not Domestic Law'

   <listed> Empty </listed> [0..1]
   'An obligation and deliverable obligation characteristic. Indicates whether or not the obligation is quoted, listed or ordinarily purchased and sold on
   an exchange. ISDA 2003 Term: Listed'

   <notContingent> Empty </notContingent> [0..1]
   'A deliverable obligation characteristic. In essence Not Contingent means the repayment of principal cannot be dependant on a formula/index, i.e. to
   prevent the risk of being delivered an instrument that may never pay any element of principal, and to ensure that the obligation is interest bearing (on
   a regular schedule). ISDA 2003 Term: Not Contingent'

   <notDomesticIssuance> Empty </notDomesticIssuance> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation other than an obligation that was intended to be offered for sale primarily in
   the domestic market of the relevant Reference Entity. This specifies that the obligation must be an internationally recognized bond. ISDA 2003 Term: Not
   Domestic Issuance'

   <assignableLoan> PCDeliverableObligationCharac </assignableLoan> [0..1]
   'A deliverable obligation characteristic. A loan that is freely assignable to a bank or financial institution without the consent of the Reference Entity
   or the guarantor, if any, of the loan (or the consent of the applicable borrower if a Reference Entity is guaranteeing the loan) or any agent. ISDA 2003
   Term: Assignable Loan'

   <consentRequiredLoan> PCDeliverableObligationCharac </consentRequiredLoan> [0..1]
   'A deliverable obligation characteristic. A loan that is capable of being assigned with the consent of the Reference Entity or the guarantor, if any, of
   the loan or any agent. ISDA 2003 Term: Consent Required Loan'

   <directLoanParticipation> LoanParticipation </directLoanParticipation> [0..1]
   'A deliverable obligation characteristic. A loan with a participation agreement whereby the buyer is capable of creating, or procuring the creation of, a
   contractual right in favour of the seller that provides the seller with recourse to the participation seller for a specified share in any payments due
   under the relevant loan which are received by the participation seller. ISDA 2003 Term: Direct Loan Participation'
```

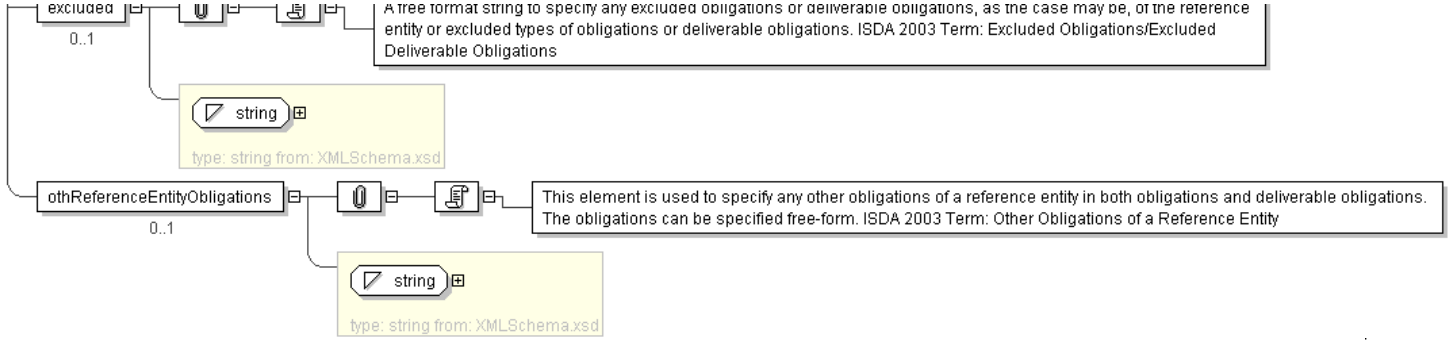
```
<transferable> Empty </transferable> [0..1]
'A deliverable obligation characteristic. An obligation that is transferable to institutional investors without any contractual, statutory or regulatory restrictions. ISDA 2003 Term: Transferable'

<maximumMaturity> Period </maximumMaturity> [0..1]
'A deliverable obligation characteristic. An obligation that has a remaining maturity from the Physical Settlement Date of not greater than the period specified. ISDA 2003 Term: Maximum Maturity'

<acceleratedOrMatured> Empty </acceleratedOrMatured> [0..1]
'A deliverable obligation characteristic. An obligation at time of default is due to mature and due to be repaid, or as a result of downgrade/bankruptcy is due to be repaid as a result of an acceleration clause. ISDA 2003 Term: Accelerated or Matured'

<notBearer> Empty </notBearer> [0..1]
'A deliverable obligation characteristic. Any obligation that is not a bearer instrument. This applies to Bonds only and is meant to avoid tax, fraud and security/delivery provisions that can potentially be associated with Bearer Bonds. ISDA 2003 Term: Not Bearer'

Start Choice [0..1]
   <fullFaithAndCreditObLiability> Empty </fullFaithAndCreditObLiability> [1]
   'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Full Faith and Credit Obligation Liability'

   <generalFundObligationLiability> Empty </generalFundObligationLiability> [1]
   'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: General Fund Obligation Liability'

   <revenueObligationLiability> Empty </revenueObligationLiability> [1]
   'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Revenue Obligation Liability'

End Choice
<indirectLoanParticipation> LoanParticipation </indirectLoanParticipation> [0..1]
'ISDA 1999 Term: Indirect Loan Participation. NOTE: Only applicable as a deliverable obligation under ISDA Credit 1999.'

<excluded> xsd:string </excluded> [0..1]
'A free format string to specify any excluded obligations or deliverable obligations, as the case may be, of the reference entity or excluded types of obligations or deliverable obligations. ISDA 2003 Term: Excluded Obligations/Excluded Deliverable Obligations'

<othReferenceEntityObligations> xsd:string </othReferenceEntityObligations> [0..1]
'This element is used to specify any other obligations of a reference entity in both obligations and deliverable obligations. The obligations can be specified free-form. ISDA 2003 Term: Other Obligations of a Reference Entity'

</...>
```

Diagram




notSovereignLender
0..1
An obligation and deliverable obligation characteristic. Any obligation that is not primarily (majority) owed to a Sovereign or Supranational Organization. ISDA 2003 Term: Not Sovereign Lender
Empty
type: Empty from: fpml-shared-4-8.xsd
notDomesticCurrency
0..1
An obligation and deliverable obligation characteristic. Any obligation that is payable in any currency other than the domestic currency. Domestic currency is either the currency so specified or, if no currency is specified, the currency of (a) the reference entity, if the reference entity is a sovereign, or (b) the jurisdiction in which the relevant reference entity is organised, if the reference entity is not a sovereign. ISDA 2003 Term: Not Domestic Currency
NotDomesticCurrency
type: NotDomesticCurrency
notDomesticLaw
0..1
An obligation and deliverable obligation characteristic. If the reference entity is a Sovereign, this means any obligation that is not subject to the laws of the reference entity. If the reference entity is not a sovereign, this means any obligation that is not subject to the laws of the jurisdiction of the reference entity. ISDA 2003 Term: Not Domestic Law
Empty
type: Empty from: fpml-shared-4-8.xsd
listed
0..1
An obligation and deliverable obligation characteristic. Indicates whether or not the obligation is quoted, listed or ordinarily purchased and sold on an exchange. ISDA 2003 Term: Listed
Empty
type: Empty from: fpml-shared-4-8.xsd
notContingent
0..1
A deliverable obligation characteristic. In essence Not Contingent means the repayment of principal cannot be dependant on a formula/index, i.e. to prevent the risk of being delivered an instrument that may never pay any element of principal, and to ensure that the obligation is interest bearing (on a regular schedule). ISDA 2003 Term: Not Contingent
Empty
type: Empty from: fpml-shared-4-8.xsd
notDomesticIssuance
0..1
An obligation and deliverable obligation characteristic. Any obligation other than an obligation that was intended to be offered for sale primarily in the domestic market of the relevant Reference Entity. This specifies that the obligation must be an internationally recognized bond. ISDA 2003 Term: Not Domestic Issuance
Empty
type: Empty from: fpml-shared-4-8.xsd
assignableLoan
0..1
A deliverable obligation characteristic. A loan that is freely assignable to a bank or financial institution without the consent of the Reference Entity or the guarantor, if any, of the loan (or the consent of the applicable borrower if a Reference Entity is guaranteeing the loan) or any agent. ISDA 2003 Term: Assignable Loan
PCDeliverableObligationCharac
type: PCDeliverableObligationCharac
consentRequiredLoan
0..1
A deliverable obligation characteristic. A loan that is capable of being assigned with the consent of the Reference Entity or the guarantor, if any, of the loan or any agent. ISDA 2003 Term: Consent Required Loan
PCDeliverableObligationCharac
type: PCDeliverableObligationCharac
directLoanParticipation
0..1
A deliverable obligation characteristic. A loan with a participation agreement whereby the buyer is capable of creating, or procuring the creation of, a contractual right in favour of the seller that provides the seller with recourse to the participation


seller for a specified share in any payments due under the relevant loan which are received by the participation seller. ISDA 2003 Term: Direct Loan Participation
LoanParticipation
type: LoanParticipation
transferable
0..1
A deliverable obligation characteristic. An obligation that is transferable to institutional investors without any contractual, statutory or regulatory restrictions. ISDA 2003 Term: Transferable
Empty
type: Empty from: fpml-shared-4-8.xsd
maximumMaturity
0..1
A deliverable obligation characteristic. An obligation that has a remaining maturity from the Physical Settlement Date of not greater than the period specified. ISDA 2003 Term: Maximum Maturity
Period
type: Period from: fpml-shared-4-8.xsd
acceleratedOrMatured
0..1
A deliverable obligation characteristic. An obligation at time of default is due to mature and due to be repaid, or as a result of downgrade/bankruptcy is due to be repaid as a result of an acceleration clause. ISDA 2003 Term: Accelerated or Matured
Empty
type: Empty from: fpml-shared-4-8.xsd
notBearer
0..1
A deliverable obligation characteristic. Any obligation that is not a bearer instrument. This applies to Bonds only and is meant to avoid tax, fraud and security/delivery provisions that can potentially be associated with Bearer Bonds. ISDA 2003 Term: Not Bearer
Empty
type: Empty from: fpml-shared-4-8.xsd
0..1
fullFaithAndCreditObLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Full Faith and Credit Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
generalFundObligationLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: General Fund Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
revenueObligationLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Revenue Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
indirectLoanParticipation
0..1
ISDA 1999 Term: Indirect Loan Participation. NOTE: Only applicable as a deliverable obligation under ISDA Credit 1999.
LoanParticipation
type: LoanParticipation



Schema Component Representation

```
<xsd:complexType name="DeliverableObligations">
  <xsd:sequence>
    <xsd:element name="accruedInterest" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="category" type=" ObligationCategoryEnum " minOccurs="0"/>
    <xsd:element name="notSubordinated" type=" Empty " minOccurs="0"/>
    <xsd:element name="specifiedCurrency" type=" SpecifiedCurrency " minOccurs="0"/>
    <xsd:element name="notSovereignLender" type=" Empty " minOccurs="0"/>
    <xsd:element name="notDomesticCurrency" type=" NotDomesticCurrency " minOccurs="0"/>
    <xsd:element name="notDomesticLaw" type=" Empty " minOccurs="0"/>
    <xsd:element name="listed" type=" Empty " minOccurs="0"/>
    <xsd:element name="notContingent" type=" Empty " minOccurs="0"/>
    <xsd:element name="notDomesticIssuance" type=" Empty " minOccurs="0"/>
    <xsd:element name="assignableLoan" type=" PCDeliverableObligationCharac " minOccurs="0"/>
    <xsd:element name="consentRequiredLoan" type=" PCDeliverableObligationCharac " minOccurs="0"/>
    <xsd:element name="directLoanParticipation" type=" LoanParticipation " minOccurs="0"/>
    <xsd:element name="transferable" type=" Empty " minOccurs="0"/>
    <xsd:element name="maximumMaturity" type=" Period " minOccurs="0"/>
    <xsd:element name="acceleratedOrMatured" type=" Empty " minOccurs="0"/>
    <xsd:element name="notBearer" type=" Empty " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="fullFaithAndCreditObLiability" type=" Empty "/>
      <xsd:element name="generalFundObligationLiability" type=" Empty "/>
      <xsd:element name="revenueObligationLiability" type=" Empty "/>
    </xsd:choice>
    <xsd:element name="indirectLoanParticipation" type=" LoanParticipation " minOccurs="0"/>
    <xsd:element name="excluded" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="othReferenceEntityObligations" type=" xsd:string " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedScheduledTerminationDate

[Table of contents]

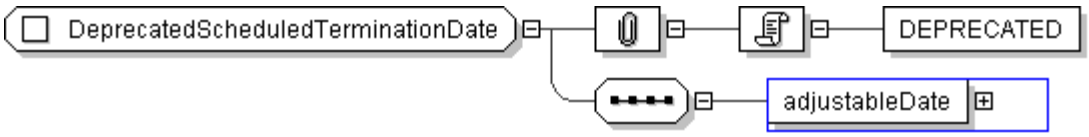
Super-types:	None
Sub-types:	None

Name	DeprecatedScheduledTerminationDate
Used by (from the same schema document)	Complex Type GeneralTerms
Abstract	no
Documentation	DEPRECATED

XML Instance Representation

```
<...>
   <adjustableDate> AdjustableDate2 </adjustableDate> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedScheduledTerminationDate" deprecated="true"
deprecatedReason="The intent is to make the scheduledTerminationDate of type
AdjustableDate2 and remove the adjustableDate node. The current container doesn't give
any value since the choice with relative date was removed.">
   <xsd:sequence>
      <xsd:element name="adjustableDate" type=" AdjustableDate2 "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EntityType

[Table of contents]

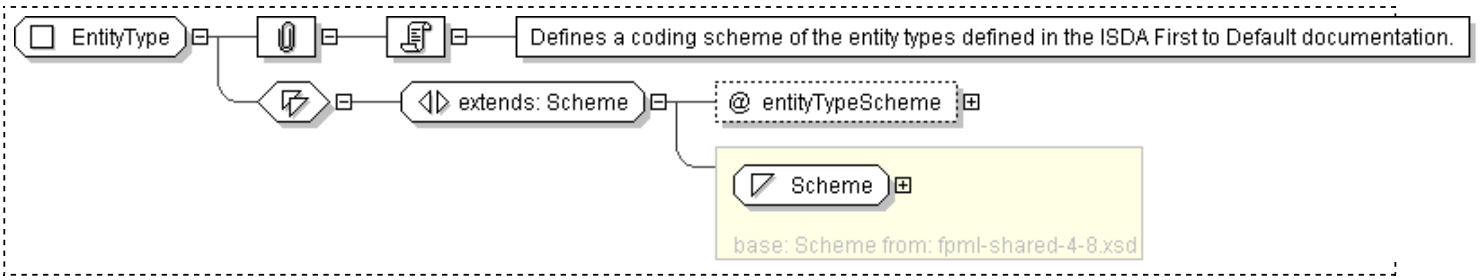
Super-types:	Scheme < EntityType (by extension)
Sub-types:	None

Name	EntityType
Used by (from the same schema document)	Complex Type ReferencePair
Abstract	no
Documentation	Defines a coding scheme of the entity types defined in the ISDA First to Default documentation.

XML Instance Representation

```
<...
 entityTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EntityType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="entityTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/entity-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FeeLeg

[Table of contents]

Super-types:	Leg < FeeLeg (by extension)
Sub-types:	None

Name	FeeLeg
Used by (from the same schema document)	Complex Type CreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialPayment> InitialPayment </initialPayment> [0..1]
```

'Specifies a single fixed payment that is payable by the payer to the receiver on the initial payment date. The fixed payment to be paid is specified in terms of a known currency amount. This element should be used for CDS Index trades and can be used for CDS trades where it is necessary to represent a payment from Seller to Buyer. For CDS trades where a payment is to be made from Buyer to Seller the feeLeg/singlePayment structure must be used.'

```
   <singlePayment> SinglePayment </singlePayment> [0..*]
```

'Specifies a single fixed amount that is payable by the buyer to the seller on the fixed rate payer payment date. The fixed amount to be paid is specified in terms of a known currency amount.'

```
   <periodicPayment> PeriodicPayment </periodicPayment> [0..1]
```

'Specifies a periodic schedule of fixed amounts that are payable by the buyer to the seller on the fixed rate payer payment dates. The fixed amount to be paid on each payment date can be specified in terms of a known currency amount or as an amount calculated on a formula basis by reference to a per annum fixed rate. The applicable business day convention and business day for adjusting any fixed rate payer payment date if it would otherwise fall on a day that is not a business day are those specified in the dateAdjustments element within the generalTerms component. ISDA 2003 Term:'

```
   <marketFixedRate> xsd:decimal </marketFixedRate> [0..1]
```

'An optional element that only has meaning in a credit index trade. This element contains the credit spread (\"fair value\") at which the trade was executed. Unlike the fixedRate of an index, the marketFixedRate varies over the life of the index depending on market conditions. The marketFixedRate is the price of the index as quoted by trading desks.'

```
   <paymentDelay> xsd:boolean </paymentDelay> [0..1]
```

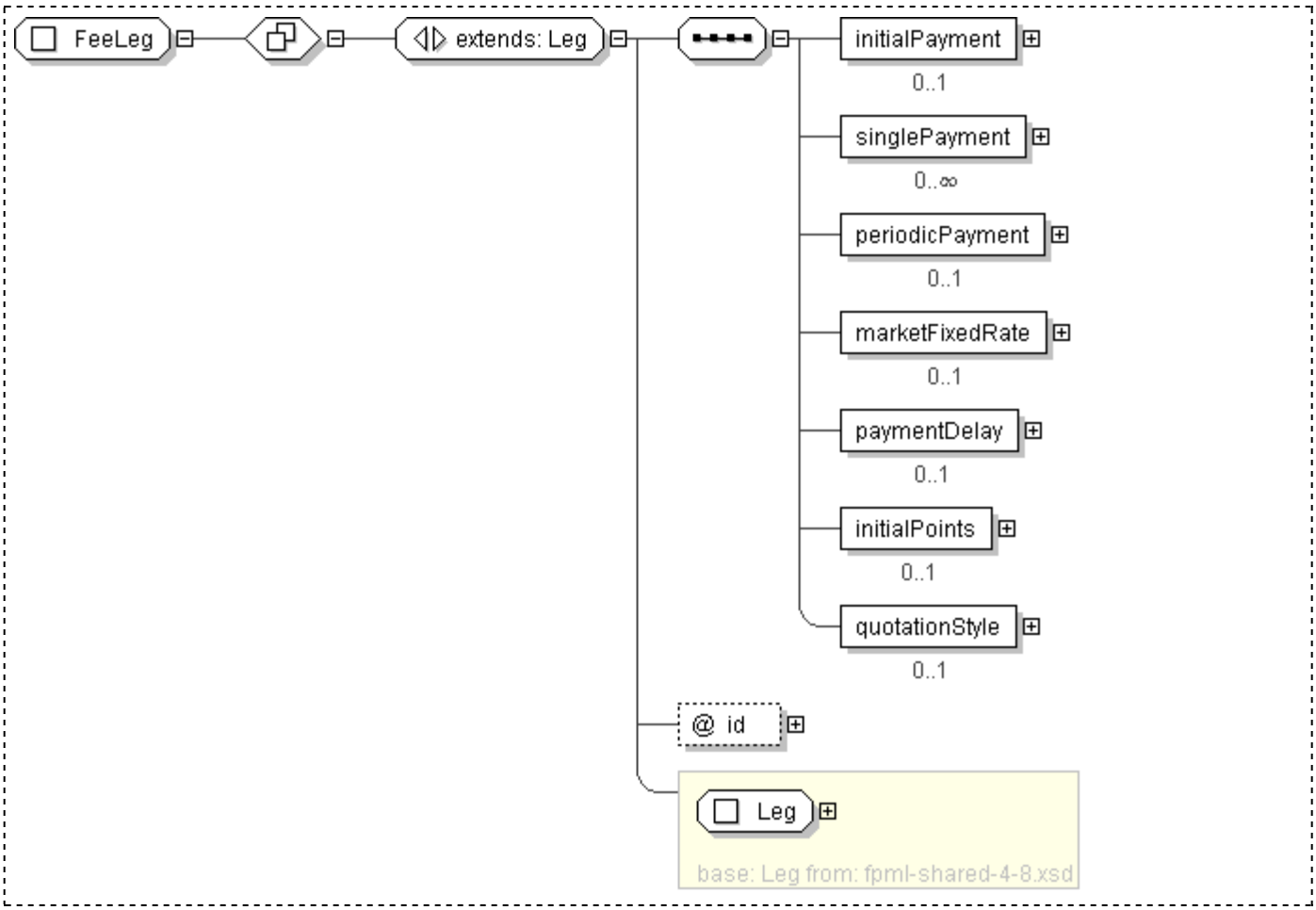
'Applicable to CDS on MBS to specify whether payment delays are applicable to the fixed Amount. RMBS typically have a payment delay of 5 days between the coupon date of the reference obligation and the payment date of the synthetic swap. CMBS do not, on the other hand, with both payment dates being on the 25th of each month.'

```
   <initialPoints> xsd:decimal </initialPoints> [0..1]
```

'An optional element that contains the up-front points expressed as a percentage of the notional. An initialPoints value of 5% would be represented as 0.05. The initialPoints element is an alternative to marketFixedRate in quoting the traded level of a trade. The initialPoints is one of the items that are factored into the initialPayment calculation and is payable by the Buyer to the Seller. Note that initialPoints and marketFixedRate may both be present in the same document when both implied values are desired.'

```
<quotationStyle> QuotationStyleEnum </quotationStyle> [0..1]
'The type of quotation that was used between the trading desks. The purpose of
this element is to indicate the actual quotation style that was used to quote
this trade which may not be apparent when both marketFixedRate and
initialPoints are included in the document. When quotationStyle is â
€˜PointsUpFrontâ€™, the initialPoints element should be populated. When
quotationStyle is â€˜TradedSpreadâ€™, the marketFixedRate element should be
populated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FeeLeg">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:element name="initialPayment" type=" InitialPayment " minOccurs="0"/>
        <xsd:element name="singlePayment" type=" SinglePayment " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="periodicPayment" type=" PeriodicPayment " minOccurs="0"/>
        <xsd:element name="marketFixedRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="paymentDelay" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="initialPoints" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="quotationStyle" type=" QuotationStyleEnum "
        minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedAmountCalculation

[Table of contents]

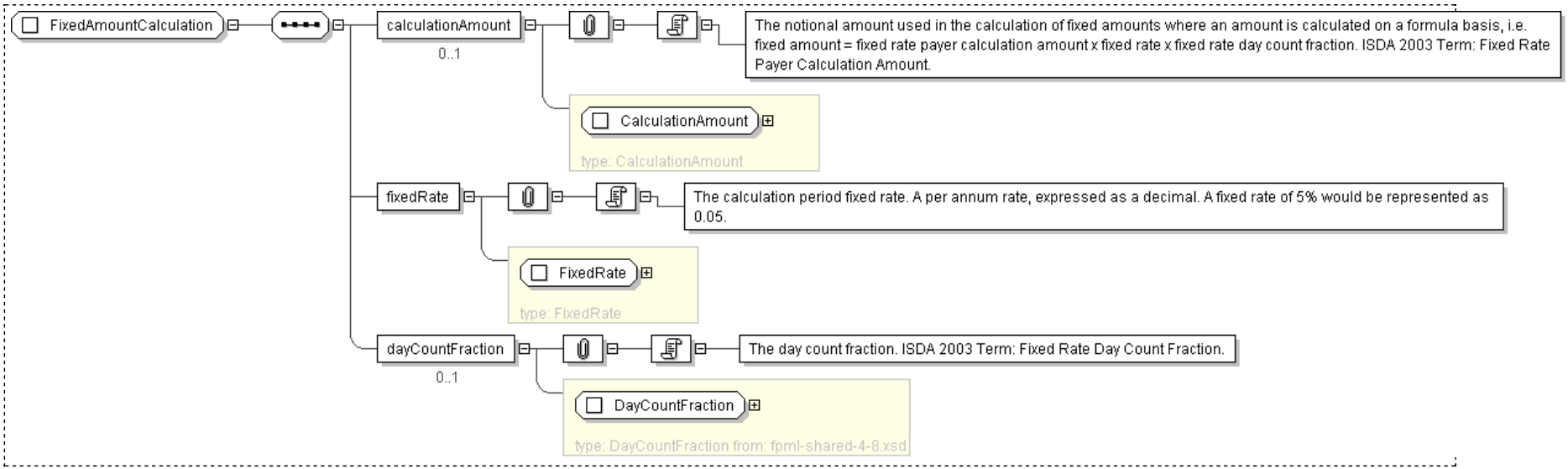
Super-types:	None
Sub-types:	None

Name	FixedAmountCalculation
Used by (from the same schema document)	Complex Type PeriodicPayment
Abstract	no

XML Instance Representation

```
<...>
  <calculationAmount> CalculationAmount </calculationAmount> [0..1]
  'The notional amount used in the calculation of fixed amounts where an amount is calculated on a formula basis, i.e. fixed amount = fixed
  rate payer calculation amount x fixed rate x fixed rate day count fraction. ISDA 2003 Term: Fixed Rate Payer Calculation Amount.'

  <fixedRate> FixedRate </fixedRate> [1]
  'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.'

  <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
  'The day count fraction. ISDA 2003 Term: Fixed Rate Day Count Fraction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedAmountCalculation">
  <xsd:sequence>
    <xsd:element name="calculationAmount" type=" CalculationAmount " minOccurs="0"/>
    <xsd:element name="fixedRate" type=" FixedRate "/>
    <xsd:element name="dayCountFraction" type=" DayCountFraction " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedRate

[Table of contents]

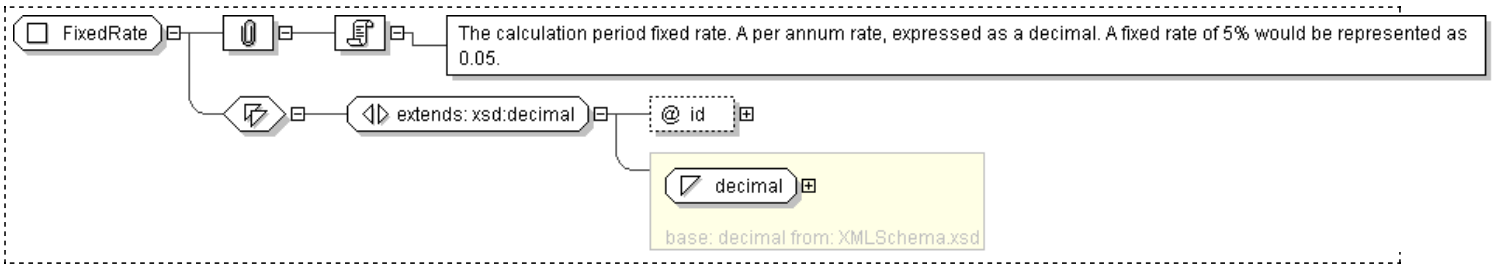
Super-types:	xsd:decimal < FixedRate (by extension)
Sub-types:	None

Name	FixedRate
Used by (from the same schema document)	Complex Type FixedAmountCalculation
Abstract	no
Documentation	The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   xsd:decimal
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedRate">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:decimal ">
      <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedRateReference

[Table of contents]

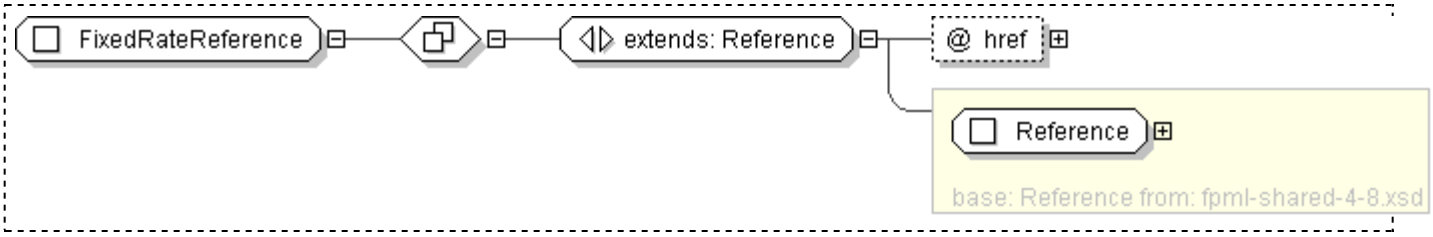
Super-types:	Reference < FixedRateReference (by extension)
Sub-types:	None

Name	FixedRateReference
Used by (from the same schema document)	Complex Type CreditOptionStrike
Abstract	no

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedRateReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="FixedRate"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingAmountEvents

[Table of contents]

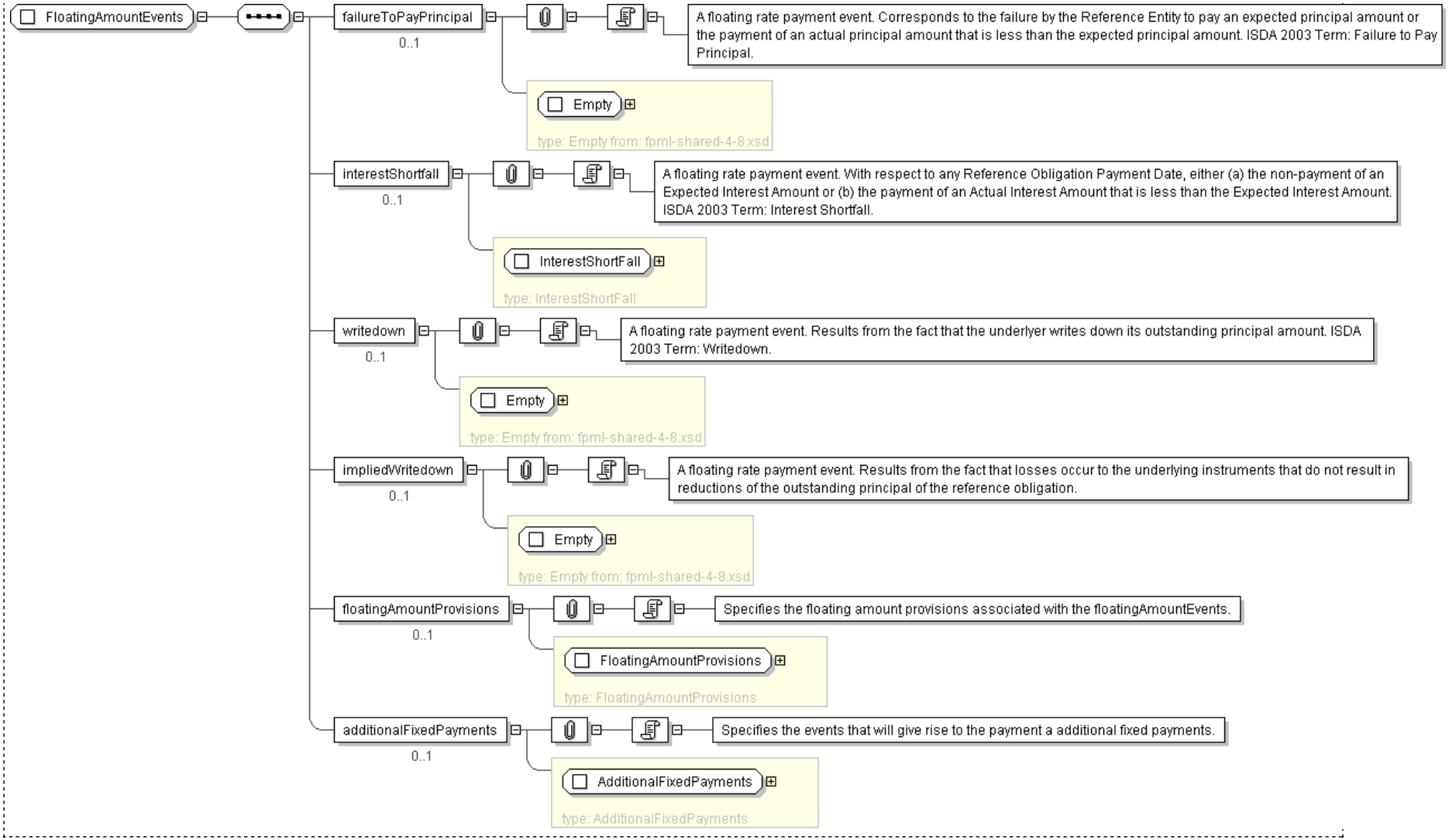
Super-types:	None
Sub-types:	None

Name	FloatingAmountEvents
Used by (from the same schema document)	Complex Type ProtectionTerms
Abstract	no

XML Instance Representation

```
<...>
   <failureToPayPrincipal> Empty </failureToPayPrincipal> [0..1]
   'A floating rate payment event. Corresponds to the failure by the Reference Entity to pay an expected principal amount or the payment of an
   actual principal amount that is less than the expected principal amount. ISDA 2003 Term: Failure to Pay Principal.'

   <interestShortfall> InterestShortFall </interestShortfall> [0..1]
   'A floating rate payment event. With respect to any Reference Obligation Payment Date, either (a) the non-payment of an Expected Interest
   Amount or (b) the payment of an Actual Interest Amount that is less than the Expected Interest Amount. ISDA 2003 Term: Interest Shortfall.'

   <writedown> Empty </writedown> [0..1]
   'A floating rate payment event. Results from the fact that the underlyer writes down its outstanding principal amount. ISDA 2003 Term:
   Writedown.'

   <impliedWritedown> Empty </impliedWritedown> [0..1]
   'A floating rate payment event. Results from the fact that losses occur to the underlying instruments that do not result in reductions of the
   outstanding principal of the reference obligation.'

   <floatingAmountProvisions> FloatingAmountProvisions </floatingAmountProvisions> [0..1]
   'Specifies the floating amount provisions associated with the floatingAmountEvents.'

   <additionalFixedPayments> AdditionalFixedPayments </additionalFixedPayments> [0..1]
   'Specifies the events that will give rise to the payment a additional fixed payments.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingAmountEvents">
  <xsd:sequence>
    <xsd:element name="failureToPayPrincipal" type=" Empty " minOccurs="0"/>
    <xsd:element name="interestShortfall" type=" InterestShortFall " minOccurs="0"/>
    <xsd:element name="writedown" type=" Empty " minOccurs="0"/>
    <xsd:element name="impliedWritedown" type=" Empty " minOccurs="0"/>
    <xsd:element name="floatingAmountProvisions" type=" FloatingAmountProvisions " minOccurs="0"/>
    <xsd:element name="additionalFixedPayments" type=" AdditionalFixedPayments " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingAmountProvisions

[Table of contents]

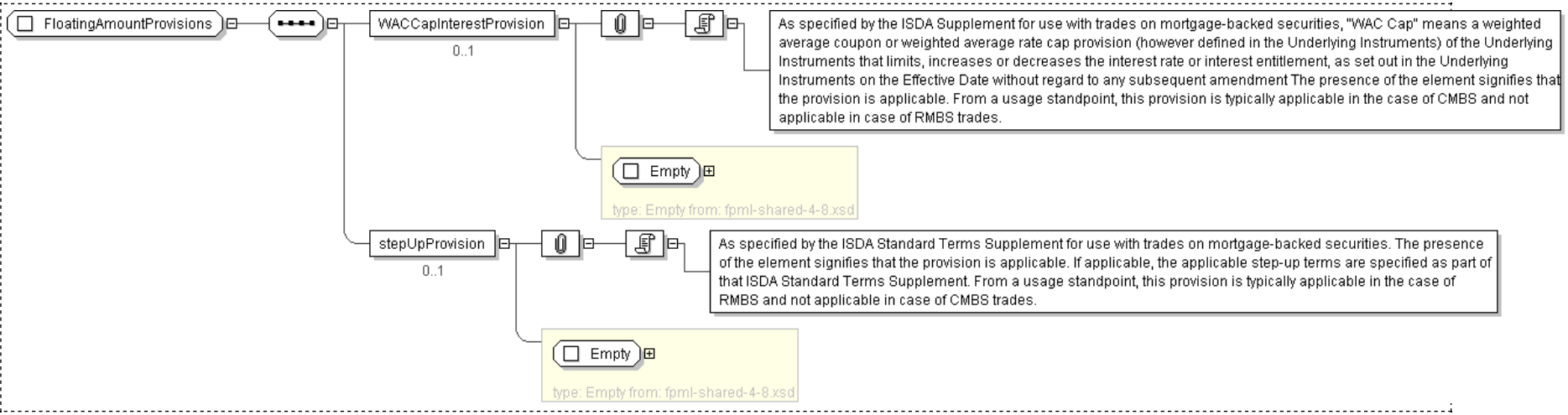
Super-types:	None
Sub-types:	None

Name	FloatingAmountProvisions
Used by (from the same schema document)	Complex Type FloatingAmountEvents
Abstract	no

XML Instance Representation

```
<...>
   <WACCapInterestProvision> Empty </WACCapInterestProvision> [0..1]
   'As specified by the ISDA Supplement for use with trades on mortgage-backed securities, \"WAC Cap\" means a weighted average coupon or weighted
   average rate cap provision (however defined in the Underlying Instruments) of the Underlying Instruments that limits, increases or decreases the
   interest rate or interest entitlement, as set out in the Underlying Instruments on the Effective Date without regard to any subsequent amendment The
   presence of the element signifies that the provision is applicable. From a usage standpoint, this provision is typically applicable in the case of
   CMBS and not applicable in case of RMBS trades.'

   <stepUpProvision> Empty </stepUpProvision> [0..1]
   'As specified by the ISDA Standard Terms Supplement for use with trades on mortgage-backed securities. The presence of the element signifies that
   the provision is applicable. If applicable, the applicable step-up terms are specified as part of that ISDA Standard Terms Supplement. From a usage
   standpoint, this provision is typically applicable in the case of RMBS and not applicable in case of CMBS trades.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingAmountProvisions">
   <xsd:sequence>
      <xsd:element name="WACCapInterestProvision" type=" Empty " minOccurs="0"/>
      <xsd:element name="stepUpProvision" type=" Empty " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GeneralTerms

[Table of contents]

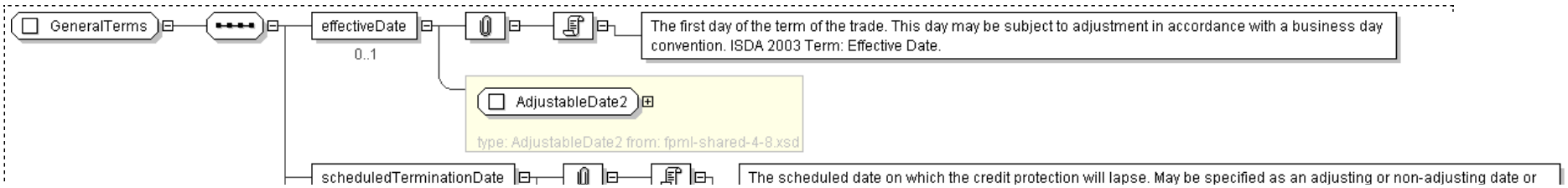
Super-types:	None
Sub-types:	None

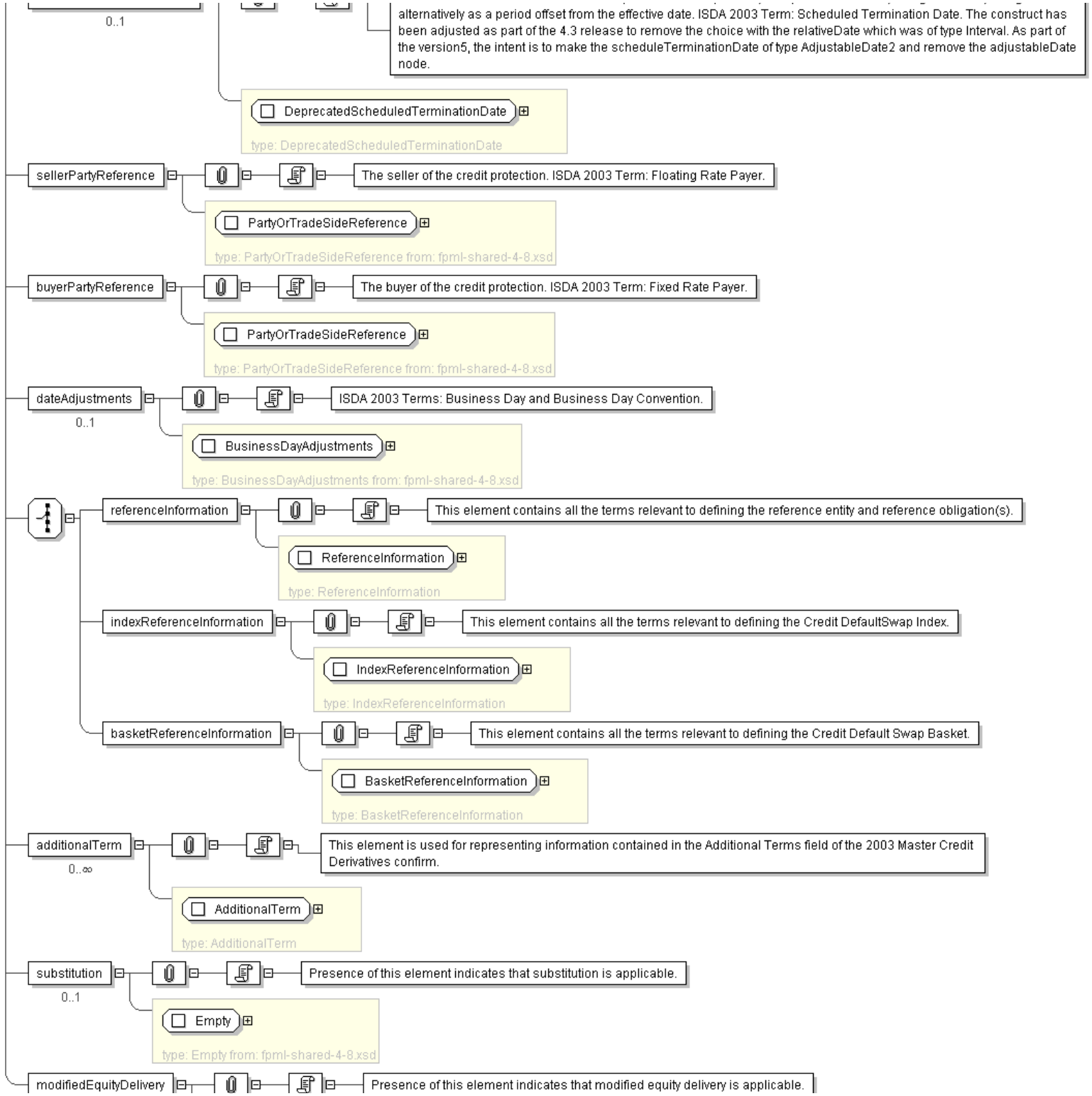
Name	GeneralTerms
Used by (from the same schema document)	Complex Type CreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...>
   <effectiveDate> AdjustableDate2 </effectiveDate> [0..1]
   'The first day of the term of the trade. This day may be subject to adjustment in accordance with a business day convention. ISDA 2003 Term:
   Effective Date.'

   <scheduledTerminationDate> DeprecatedScheduledTerminationDate </scheduledTerminationDate> [0..1]
   'The scheduled date on which the credit protection will lapse. May be specified as an adjusting or non-adjusting date or alternatively as a
   period offset from the effective date. ISDA 2003 Term: Scheduled Termination Date. The construct has been adjusted as part of the 4.3 release
   to remove the choice with the relativeDate which was of type Interval. As part of the version5, the intent is to make the
   scheduleTerminationDate of type AdjustableDate2 and remove the adjustableDate node.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'The seller of the credit protection. ISDA 2003 Term: Floating Rate Payer.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'The buyer of the credit protection. ISDA 2003 Term: Fixed Rate Payer.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [0..1]
   'ISDA 2003 Terms: Business Day and Business Day Convention.'

   Start Choice [1]
      <referenceInformation> ReferenceInformation </referenceInformation> [1]
      'This element contains all the terms relevant to defining the reference entity and reference obligation(s).'

      <indexReferenceInformation> IndexReferenceInformation </indexReferenceInformation> [1]
      'This element contains all the terms relevant to defining the Credit DefaultSwap Index.'

      <basketReferenceInformation> BasketReferenceInformation </basketReferenceInformation> [1]
      'This element contains all the terms relevant to defining the Credit Default Swap Basket.'

   End Choice
   <additionalTerm> AdditionalTerm </additionalTerm> [0..*]
   'This element is used for representing information contained in the Additional Terms field of the 2003 Master Credit Derivatives confirm.'

   <substitution> Empty </substitution> [0..1]
   'Presence of this element indicates that substitution is applicable.'

   <modifiedEquityDelivery> Empty </modifiedEquityDelivery> [0..1]
   'Presence of this element indicates that modified equity delivery is applicable.'

</...>
```

Diagram




0..1
alternatively as a period offset from the effective date. ISDA 2003 Term: Scheduled Termination Date. The construct has been adjusted as part of the 4.3 release to remove the choice with the relativeDate which was of type Interval. As part of the version5, the intent is to make the scheduleTerminationDate of type AdjustableDate2 and remove the adjustableDate node.
DeprecatedScheduledTerminationDate
type: DeprecatedScheduledTerminationDate
sellerPartyReference
The seller of the credit protection. ISDA 2003 Term: Floating Rate Payer.
PartyOrTradeSideReference
type: PartyOrTradeSideReference from: fpml-shared-4-8.xsd
buyerPartyReference
The buyer of the credit protection. ISDA 2003 Term: Fixed Rate Payer.
PartyOrTradeSideReference
type: PartyOrTradeSideReference from: fpml-shared-4-8.xsd
dateAdjustments
0..1
ISDA 2003 Terms: Business Day and Business Day Convention.
BusinessDayAdjustments
type: BusinessDayAdjustments from: fpml-shared-4-8.xsd
referenceInformation
This element contains all the terms relevant to defining the reference entity and reference obligation(s).
ReferenceInformation
type: ReferenceInformation
indexReferenceInformation
This element contains all the terms relevant to defining the Credit DefaultSwap Index.
IndexReferenceInformation
type: IndexReferenceInformation
basketReferenceInformation
This element contains all the terms relevant to defining the Credit Default Swap Basket.
BasketReferenceInformation
type: BasketReferenceInformation
additionalTerm
0..∞
This element is used for representing information contained in the Additional Terms field of the 2003 Master Credit Derivatives confirm.
AdditionalTerm
type: AdditionalTerm
substitution
0..1
Presence of this element indicates that substitution is applicable.
Empty
type: Empty from: fpml-shared-4-8.xsd
modifiedEquityDelivery
Presence of this element indicates that modified equity delivery is applicable.

Schema Component Representation

```
<xsd:complexType name="GeneralTerms">
   <xsd:sequence>
      <xsd:element name="effectiveDate" type=" AdjustableDate2 " minOccurs="0"/>
      <xsd:element name="scheduledTerminationDate" type=" DeprecatedScheduledTerminationDate " minOccurs="0"/>
      <xsd:element name="sellerPartyReference" type=" PartyOrTradeSideReference "/>
      <xsd:element name="buyerPartyReference" type=" PartyOrTradeSideReference "/>
      <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments " minOccurs="0"/>
      <xsd:choice>
         <xsd:element name="referenceInformation" type=" ReferenceInformation "/>
         <xsd:element name="indexReferenceInformation" type=" IndexReferenceInformation "/>
         <xsd:element name="basketReferenceInformation" type=" BasketReferenceInformation "/>
      </xsd:choice>
      <xsd:element name="additionalTerm" type=" AdditionalTerm " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="substitution" type=" Empty " minOccurs="0"/>
      <xsd:element name="modifiedEquityDelivery" type=" Empty " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexAnnexSource

[Table of contents]

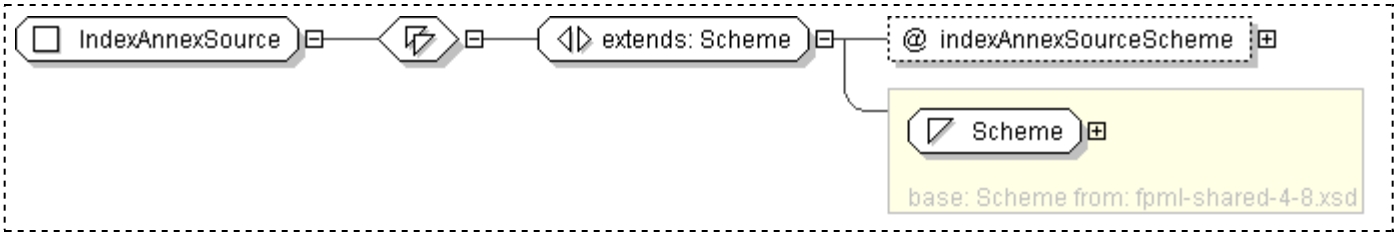
Super-types:	Scheme < IndexAnnexSource (by extension)
Sub-types:	None

Name	IndexAnnexSource
Used by (from the same schema document)	Complex Type IndexReferenceInformation
Abstract	no

XML Instance Representation

```
<...
 indexAnnexSourceScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexAnnexSource">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="indexAnnexSourceScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/cdx-index-annex-source"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexId

[Table of contents]

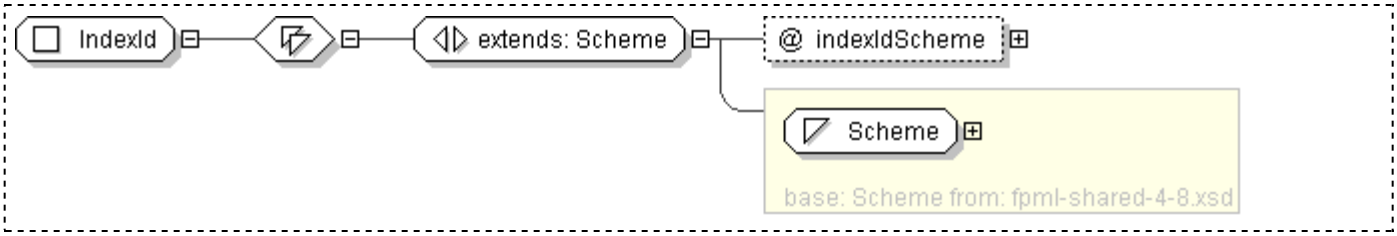
Super-types:	Scheme < IndexId (by extension)
Sub-types:	None

Name	IndexId
Used by (from the same schema document)	Complex Type IndexReferenceInformation , Complex Type IndexReferenceInformation
Abstract	no

XML Instance Representation

```
<...
 indexIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="indexIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexName

[Table of contents]

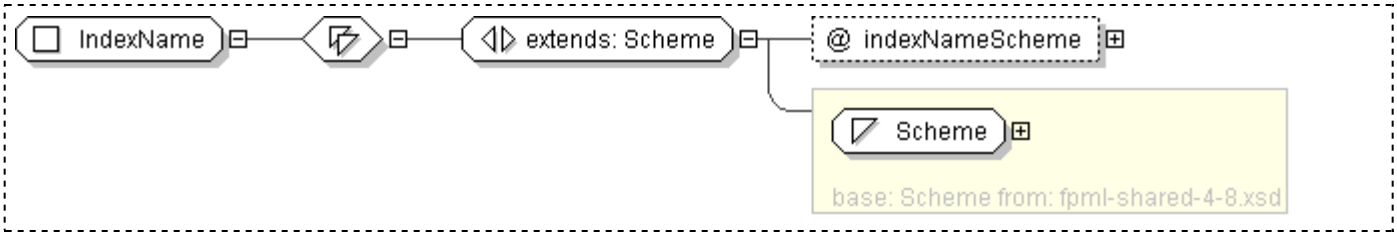
Super-types:	Scheme < IndexName (by extension)
Sub-types:	None

Name	IndexName
Used by (from the same schema document)	Complex Type IndexReferenceInformation
Abstract	no

XML Instance Representation

```
<...
 indexNameScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexName">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="indexNameScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexReferenceInformation

[Table of contents]

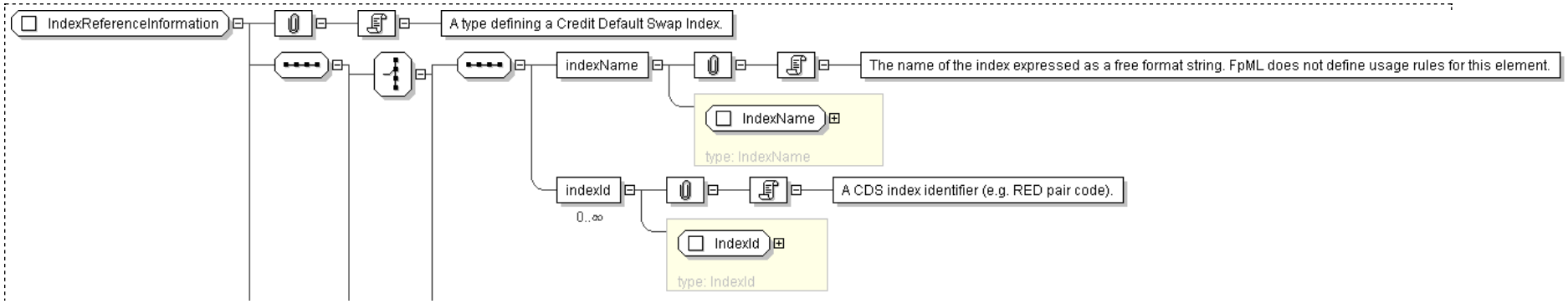
Super-types:	None
Sub-types:	None

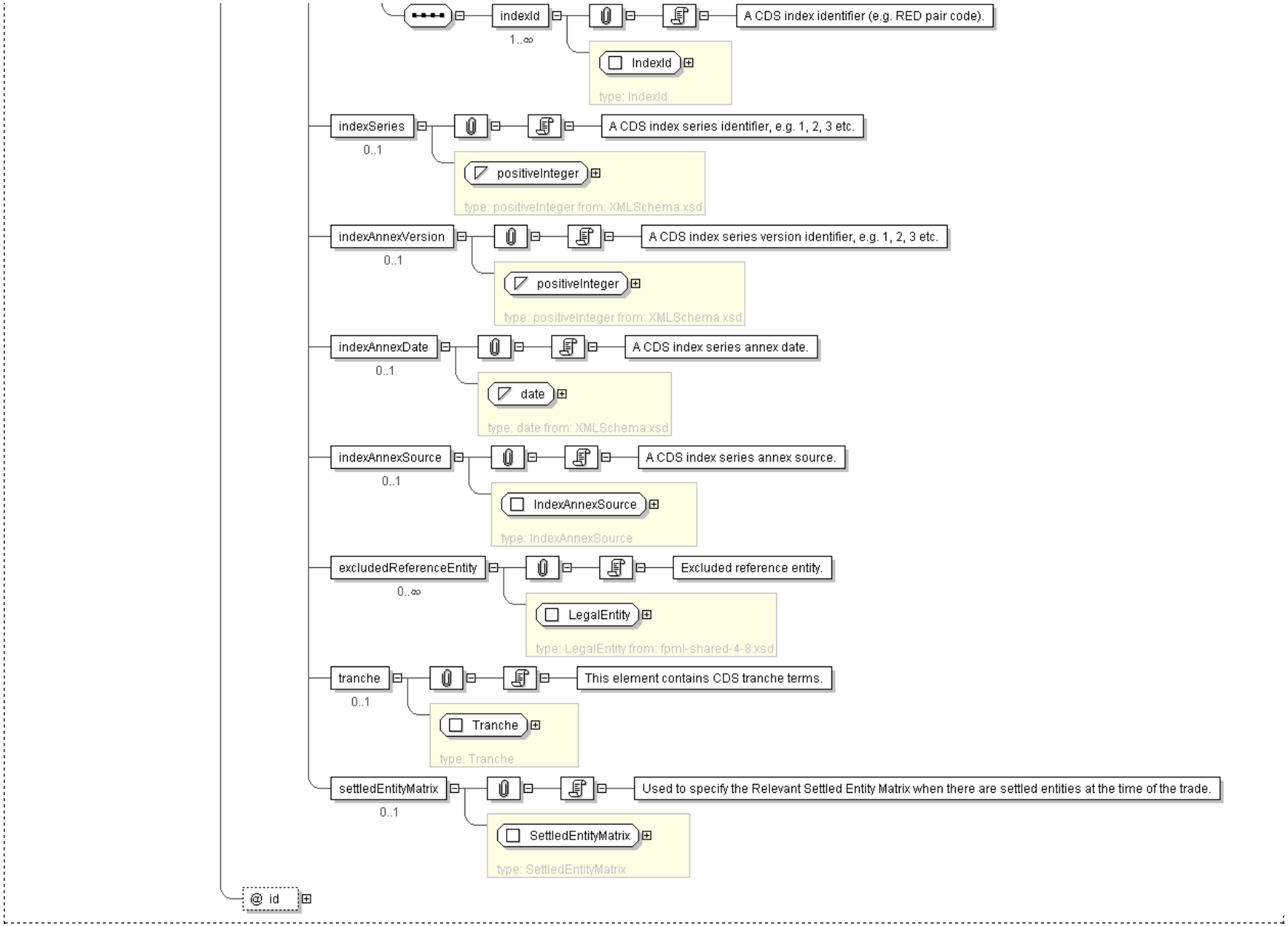
Name	IndexReferenceInformation
Used by (from the same schema document)	Complex Type GeneralTerms
Abstract	no
Documentation	A type defining a Credit Default Swap Index.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <indexName> IndexName </indexName> [1]
      'The name of the index expressed as a free format string. FpML does not define usage rules for this element.'

      <indexId> IndexId </indexId> [0..*]
      'A CDS index identifier (e.g. RED pair code).'

      <indexId> IndexId </indexId> [1..*]
      'A CDS index identifier (e.g. RED pair code).'

   End Choice
   <indexSeries> xsd:positiveInteger </indexSeries> [0..1]
   'A CDS index series identifier, e.g. 1, 2, 3 etc.'

   <indexAnnexVersion> xsd:positiveInteger </indexAnnexVersion> [0..1]
   'A CDS index series version identifier, e.g. 1, 2, 3 etc.'

   <indexAnnexDate> xsd:date </indexAnnexDate> [0..1]
   'A CDS index series annex date.'

   <indexAnnexSource> IndexAnnexSource </indexAnnexSource> [0..1]
   'A CDS index series annex source.'

   <excludedReferenceEntity> LegalEntity </excludedReferenceEntity> [0..*]
   'Excluded reference entity.'

   <tranche> Tranche </tranche> [0..1]
   'This element contains CDS tranche terms.'

   <settledEntityMatrix> SettledEntityMatrix </settledEntityMatrix> [0..1]
   'Used to specify the Relevant Settled Entity Matrix when there are settled entities at the time of the trade.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="IndexReferenceInformation">
  <xsd:sequence>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="indexName" type=" IndexName "/>
        <xsd:element name="indexId" type=" IndexId " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="indexId" type=" IndexId " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:choice>
    <xsd:element name="indexSeries" type=" xsd:positiveInteger " minOccurs="0"/>
    <xsd:element name="indexAnnexVersion" type=" xsd:positiveInteger " minOccurs="0"/>
```

```
      <xsd:element name="indexAnnexDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="indexAnnexSource" type=" IndexAnnexSource " minOccurs="0"/>
      <xsd:element name="excludedReferenceEntity" type=" LegalEntity " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="tranche" type=" Tranche " minOccurs="0"/>
      <xsd:element name="settledEntityMatrix" type=" SettledEntityMatrix " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InitialPayment

[Table of contents]

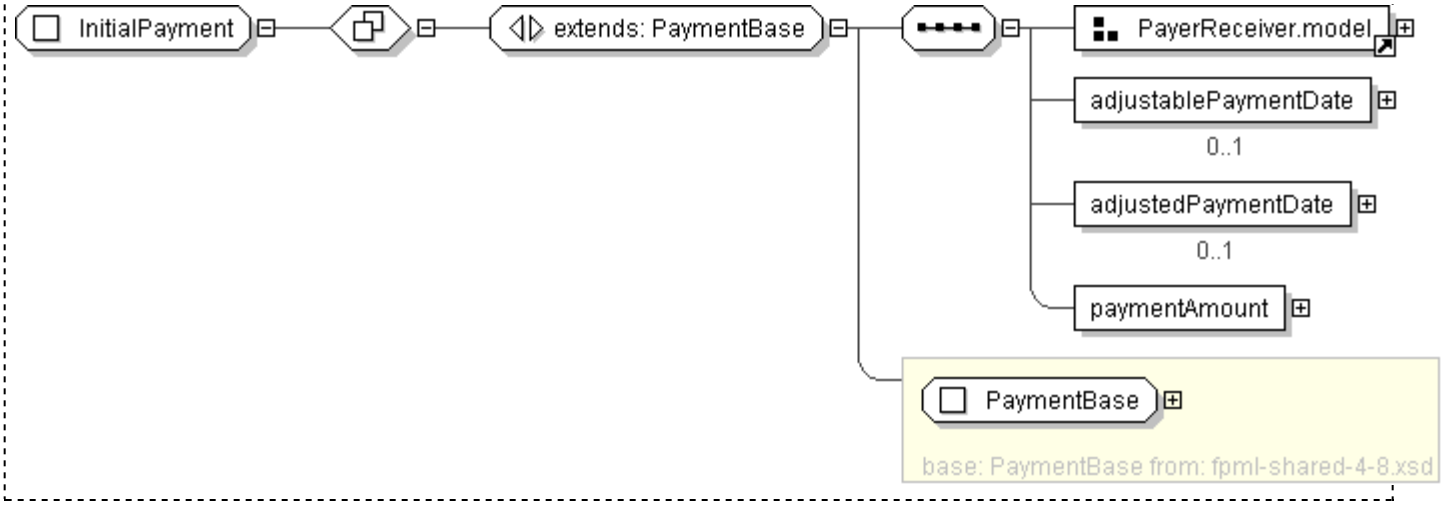
Super-types:	PaymentBase < InitialPayment (by extension)
Sub-types:	None

Name	InitialPayment
Used by (from the same schema document)	Complex Type FeeLeg
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <adjustablePaymentDate> xsd:date </adjustablePaymentDate> [0..1]
   'A fixed payment date that shall be subject to adjustment in accordance with
   the applicable business day convention if it would otherwise fall on a day that
   is not a business day. The applicable business day convention and business day
   are those specified in the dateAdjustments element within the generalTerms
   component.'

   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [0..1]
   'The adjusted payment date. This date should already be adjusted for any
   applicable business day convention. This component is not intended for use in
   trade confirmation but may be specified to allow the fee structure to also
   serve as a cashflow type component.'

   <paymentAmount> Money </paymentAmount> [1]
   'A fixed payment amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InitialPayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="adjustablePaymentDate" type=" xsd:date " minOccurs="0"/>
            <xsd:element name="adjustedPaymentDate" type=" xsd:date " minOccurs="0"/>
            <xsd:element name="paymentAmount" type=" Money "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestShortFall

[Table of contents]

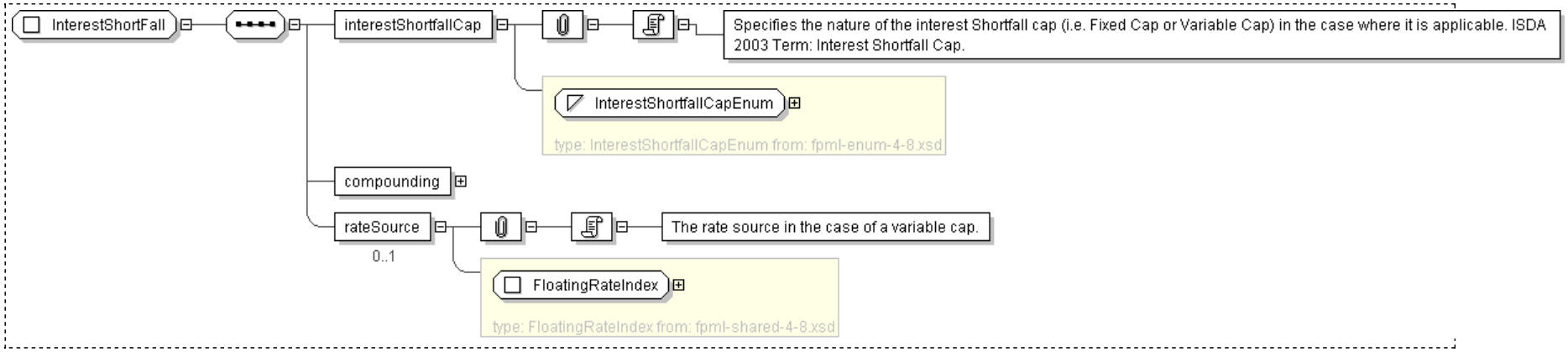
Super-types:	None
Sub-types:	None

Name	InterestShortFall
Used by (from the same schema document)	Complex Type FloatingAmountEvents
Abstract	no

XML Instance Representation

```
<...>
  <interestShortfallCap> InterestShortfallCapEnum </interestShortfallCap> [1]
  'Specifies the nature of the interest Shortfall cap (i.e. Fixed Cap or Variable Cap) in the case where it is applicable. ISDA 2003 Term:
  Interest Shortfall Cap.'

  <compounding> xsd:boolean </compounding> [1]
  <rateSource> FloatingRateIndex </rateSource> [0..1]
  'The rate source in the case of a variable cap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestShortFall">
  <xsd:sequence>
    <xsd:element name="interestShortfallCap" type=" InterestShortfallCapEnum "/>
    <xsd:element name="compounding" type=" xsd:boolean "/>
    <xsd:element name="rateSource" type=" FloatingRateIndex " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanParticipation

[Table of contents]

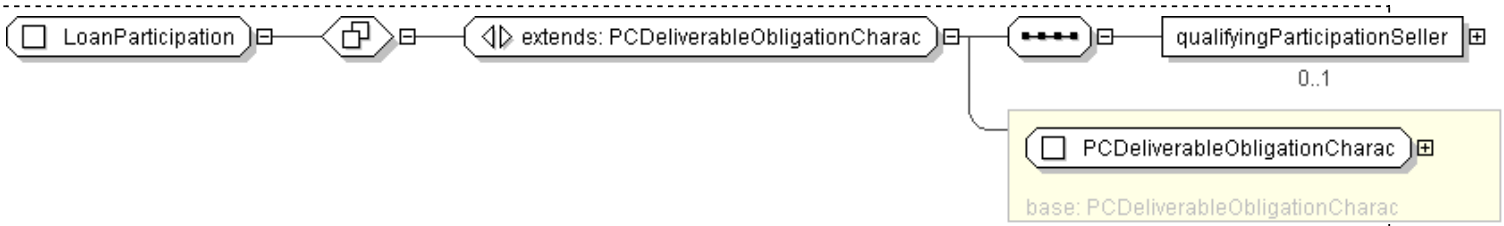
Super-types:	PCDeliverableObligationCharac < LoanParticipation (by extension)
Sub-types:	None

Name	LoanParticipation
Used by (from the same schema document)	Complex Type DeliverableObligations , Complex Type DeliverableObligations
Abstract	no

XML Instance Representation

```
<...>
  <partialCashSettlement> Empty </partialCashSettlement> [0..1]
  'Specifies whether either \'Partial Cash Settlement of Assignable Loans\', \'Partial Cash
  Settlement of Consent Required Loans\' or \'Partial Cash Settlement of Participations\' is
  applicable. If this element is specified and Assignable Loan is a Deliverable Obligation
  Chracteristic, any Assignable Loan that is deliverable, but where a non-receipt of Consent
  by the Physical Settlement Date has occurred, the Loan can be cash settled rather than
  physically delivered. If this element is specified and Consent Required Loan is a
  Deliverable Obligation Characterisitc, any Consent Required Loan that is deliverable, but
  where a non-receipt of Consent by the Physical Settlement Date has occurred, the Loan can
  be cash settled rather than physically delivered. If this element is specified and Direct
  Loan Participation is a Deliverable Obligation Characterisitic, any Participation that is
  deliverable, but where this participation has not been effected (has not come into effect)
  by the Physical Settlement Date, the participation can be cash settled rather than
  physically delivered.'

  <qualifyingParticipationSeller> xsd:string </qualifyingParticipationSeller> [0..1]
  'If Direct Loan Participation is specified as a deliverable obligation characteristic,
  this specifies any requirements for the Qualifying Participation Seller. The requirements
  may be listed free-form. ISDA 2003 Term: Qualifying Participation Seller'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LoanParticipation">
  <xsd:complexContent>
    <xsd:extension base=" PCDeliverableObligationCharac ">
      <xsd:sequence>
        <xsd:element name="qualifyingParticipationSeller" type=" xsd:string " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MatrixSource

[Table of contents]

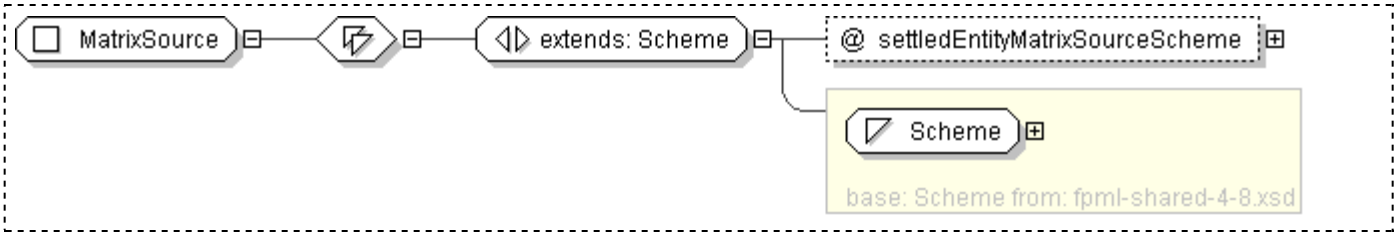
Super-types:	Scheme < MatrixSource (by extension)
Sub-types:	None

Name	MatrixSource
Used by (from the same schema document)	Complex Type SettledEntityMatrix
Abstract	no

XML Instance Representation

```
<...
 settledEntityMatrixSourceScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MatrixSource">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="settledEntityMatrixSourceScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/settled-entity-matrix-source"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MultipleValuationDates

[Table of contents]

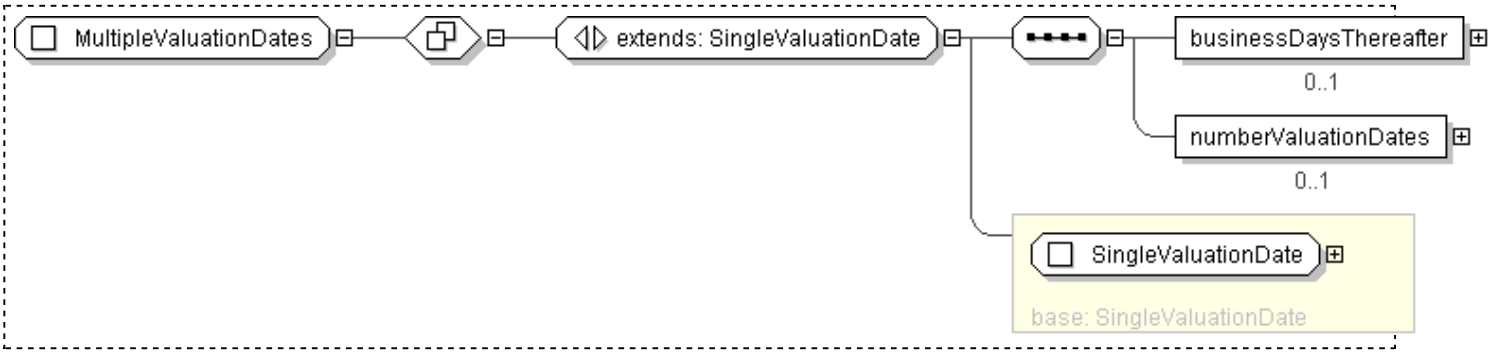
Super-types:	SingleValuationDate < MultipleValuationDates (by extension)
Sub-types:	None

Name	MultipleValuationDates
Used by (from the same schema document)	Complex Type ValuationDate
Abstract	no

XML Instance Representation

```
<...>
   <businessDays> xsd:nonNegativeInteger </businessDays> [0..1]
   'A number of business days. Its precise meaning is dependant on the context in which
   this element is used. ISDA 2003 Term: Business Day'

   <businessDaysThereafter> xsd:positiveInteger </businessDaysThereafter> [0..1]
   'The number of business days between successive valuation dates when multiple
   valuation dates are applicable for cash settlement. ISDA 2003 Term: Business Days
   thereafter'

   <numberValuationDates> xsd:positiveInteger </numberValuationDates> [0..1]
   'Where multiple valuation dates are specified as being applicable for cash
   settlement, this element specifies (a) the number of applicable valuation dates, and
   (b) the number of business days after satisfaction of all conditions to settlement
   when the first such valuation date occurs, and (c) the number of business days
   thereafter of each successive valuation date. ISDA 2003 Term: Multiple Valuation
   Dates'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MultipleValuationDates">
   <xsd:complexContent>
      <xsd:extension base=" SingleValuationDate ">
         <xsd:sequence>
            <xsd:element name="businessDaysThereafter" type=" xsd:positiveInteger "
            minOccurs="0"/>
            <xsd:element name="numberValuationDates" type=" xsd:positiveInteger "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotDomesticCurrency

[Table of contents]

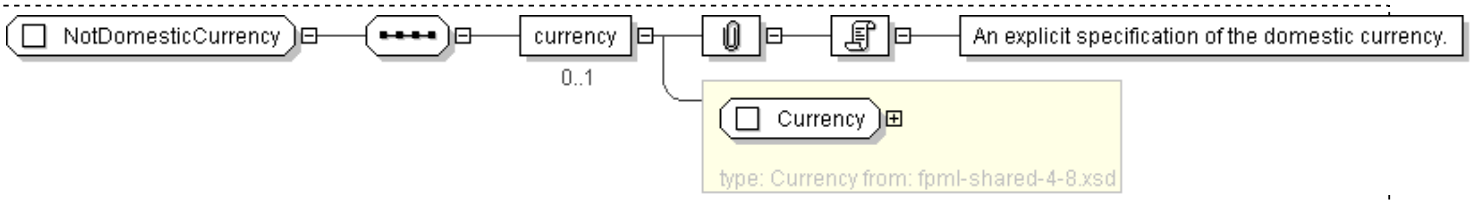
Super-types:	None
Sub-types:	None

Name	NotDomesticCurrency
Used by (from the same schema document)	Complex Type DeliverableObligations , Complex Type Obligations
Abstract	no

XML Instance Representation

```
<...>
   <currency> Currency </currency> [0..1]
   'An explicit specification of the domestic currency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotDomesticCurrency">
   <xsd:sequence>
      <xsd:element name="currency" type=" Currency " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Obligations

[Table of contents]

Super-types:	None
Sub-types:	None

Name	Obligations
Used by (from the same schema document)	Complex Type ProtectionTerms
Abstract	no

XML Instance Representation

```
<...>
   <category> ObligationCategoryEnum </category> [1]
   'Used in both obligations and deliverable obligations to represent a class or type of securities which apply. ISDA 2003 Term: Obligation
   Category/Deliverable Obligation Category'

   <notSubordinated> Empty </notSubordinated> [0..1]
   'An obligation and deliverable obligation characteristic. An obligation that ranks at least equal with the most senior Reference Obligation in
   priority of payment or, if no Reference Obligation is specified in the related Confirmation, the obligations of the Reference Entity that are
   senior. ISDA 2003 Term: Not Subordinated'

   <specifiedCurrency> SpecifiedCurrency </specifiedCurrency> [0..1]
   'An obligation and deliverable obligation characteristic. The currency or currencies in which an obligation or deliverable obligation must be
   payable. ISDA 2003 Term: Specified Currency'

   <notSovereignLender> Empty </notSovereignLender> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation that is not primarily (majority) owed to a Sovereign or Supranational
   Organization. ISDA 2003 Term: Not Sovereign Lender'

   <notDomesticCurrency> NotDomesticCurrency </notDomesticCurrency> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation that is payable in any currency other than the domestic currency.
   Domestic currency is either the currency so specified or, if no currency is specified, the currency of (a) the reference entity, if the reference
   entity is a sovereign, or (b) the jurisdiction in which the relevant reference entity is organised, if the reference entity is not a sovereign.
   ISDA 2003 Term: Not Domestic Currency'

   <notDomesticLaw> Empty </notDomesticLaw> [0..1]
   'An obligation and deliverable obligation characteristic. If the reference entity is a Sovereign, this means any obligation that is not subject to
   the laws of the reference entity. If the reference entity is not a sovereign, this means any obligation that is not subject to the laws of the
   jurisdiction of the reference entity. ISDA 2003 Term: Not Domestic Law'

   <listed> Empty </listed> [0..1]
   'An obligation and deliverable obligation characteristic. Indicates whether or not the obligation is quoted, listed or ordinarily purchased and
   sold on an exchange. ISDA 2003 Term: Listed'

   <notDomesticIssuance> Empty </notDomesticIssuance> [0..1]
   'An obligation and deliverable obligation characteristic. Any obligation other than an obligation that was intended to be offered for sale
   primarily in the domestic market of the relevant Reference Entity. This specifies that the obligation must be an internationally recognized bond.
   ISDA 2003 Term: Not Domestic Issuance'

   Start Choice [0..1]
      <fullFaithAndCreditObLiability> Empty </fullFaithAndCreditObLiability> [1]
      'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference
      Entity. ISDA 2003 Term: Full Faith and Credit Obligation Liability'

      <generalFundObligationLiability> Empty </generalFundObligationLiability> [1]
      'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference
      Entity. ISDA 2003 Term: General Fund Obligation Liability'

      <revenueObligationLiability> Empty </revenueObligationLiability> [1]
      'An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference
      Entity. ISDA 2003 Term: Revenue Obligation Liability'

   End Choice
   <notContingent> Empty </notContingent> [0..1]
   'NOTE: Only allowed as an obligation charcteristic under ISDA Credit 1999. In essence Not Contingent means the repayment of principal cannot be
   dependant on a formula/index, i.e. to prevent the risk of being delivered an instrument that may never pay any element of principal, and to ensure
   that the obligation is interest bearing (on a regular schedule). ISDA 2003 Term: Not Contingent'

   <excluded> xsd:string </excluded> [0..1]
```

'A free format string to specify any excluded obligations or deliverable obligations, as the case may be, of the reference entity or excluded types of obligations or deliverable obligations. ISDA 2003 Term: Excluded Obligations/Excluded Deliverable Obligations'

<othReferenceEntityObligations> xsd:string </othReferenceEntityObligations> [0..1]

'This element is used to specify any other obligations of a reference entity in both obligations and deliverable obligations. The obligations can be specified free-form. ISDA 2003 Term: Other Obligations of a Reference Entity'

<designatedPriority> Lien </designatedPriority> [0..1]

'Applies to Loan CDS, to indicate what lien level is appropriate for a deliverable obligation. Applies to European Loan CDS, to indicate the Ranking of the obligation. Example: a 2nd lien Loan CDS would imply that the deliverable obligations are 1st or 2nd lien loans.'

<cashSettlementOnly> Empty </cashSettlementOnly> [0..1]

'An obligation and deliverable obligation characteristic. Defined in the ISDA published Standard Terms Supplement for use with CDS Transactions on Leveraged Loans. ISDA 2003 Term: Cash Settlement Only.'

<deliveryOfCommitments> Empty </deliveryOfCommitments> [0..1]

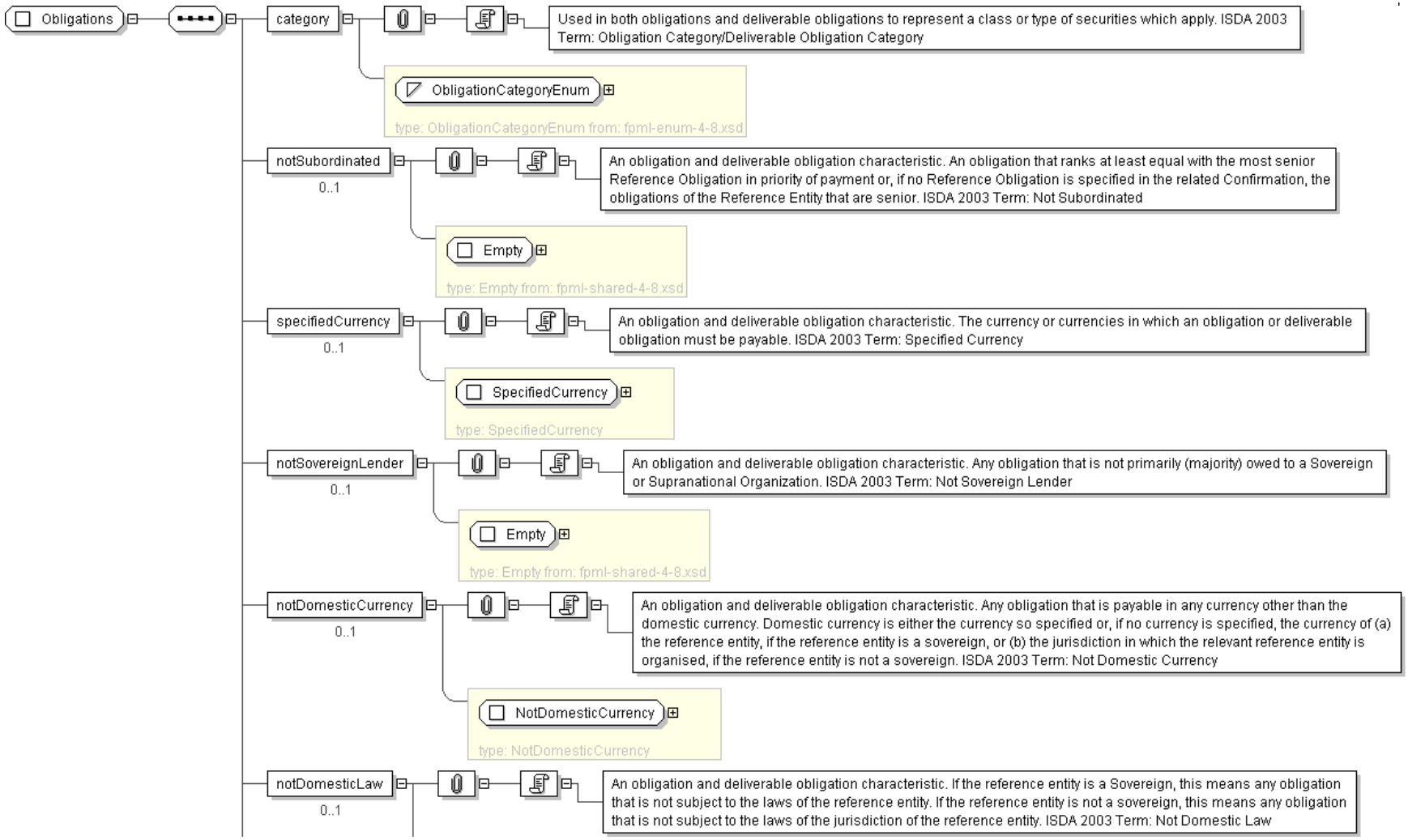
'An obligation and deliverable obligation characteristic. Defined in the ISDA published Standard Terms Supplement for use with CDS Transactions on Leveraged Loans. ISDA 2003 Term: Delivery of Commitments.'

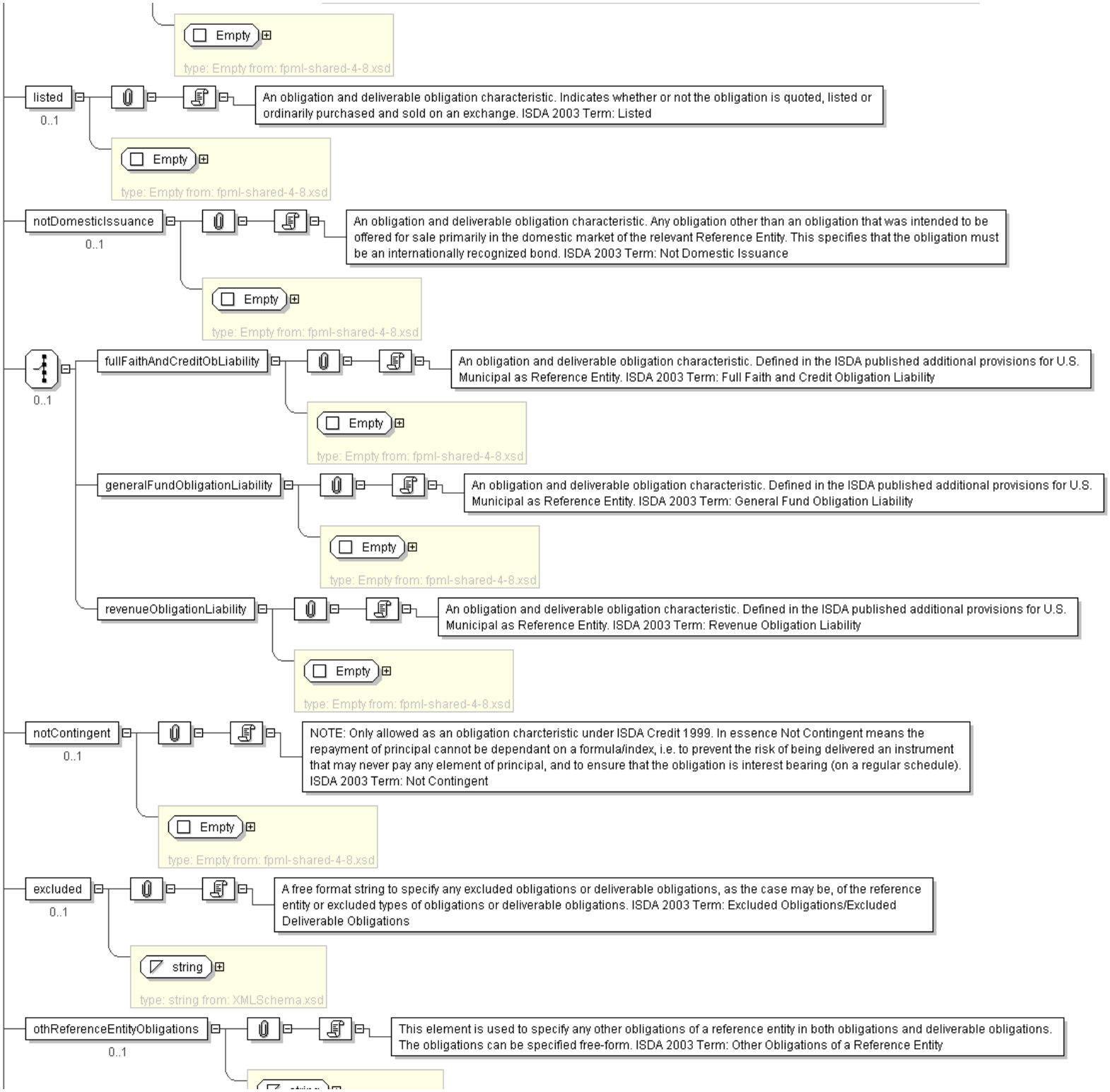
<continuity> Empty </continuity> [0..1]

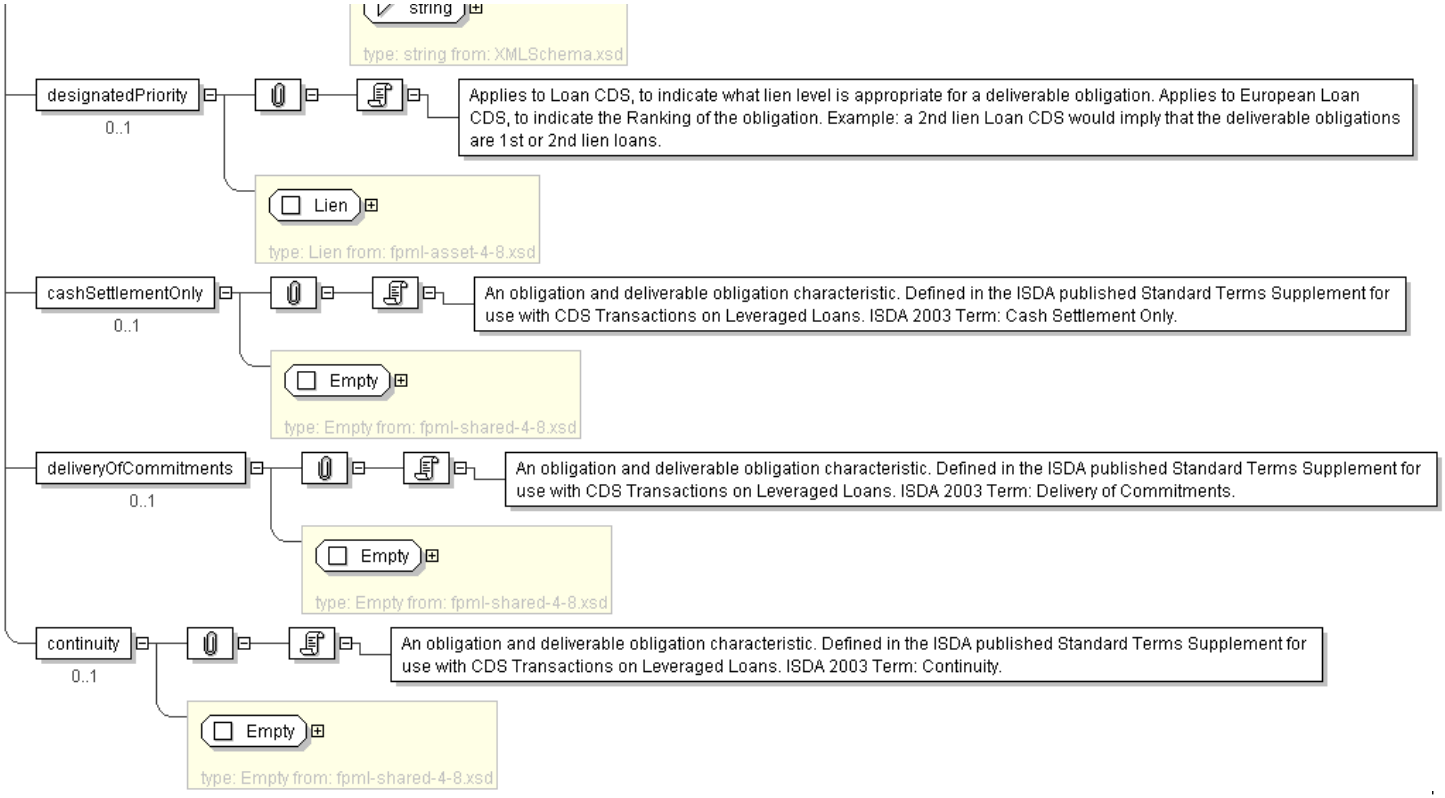
'An obligation and deliverable obligation characteristic. Defined in the ISDA published Standard Terms Supplement for use with CDS Transactions on Leveraged Loans. ISDA 2003 Term: Continuity.'

</...>

Diagram




Empty
type: Empty from: fpml-shared-4-8.xsd
listed
0..1
An obligation and deliverable obligation characteristic. Indicates whether or not the obligation is quoted, listed or ordinarily purchased and sold on an exchange. ISDA 2003 Term: Listed
Empty
type: Empty from: fpml-shared-4-8.xsd
notDomesticIssuance
0..1
An obligation and deliverable obligation characteristic. Any obligation other than an obligation that was intended to be offered for sale primarily in the domestic market of the relevant Reference Entity. This specifies that the obligation must be an internationally recognized bond. ISDA 2003 Term: Not Domestic Issuance
Empty
type: Empty from: fpml-shared-4-8.xsd
0..1
fullFaithAndCreditObLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Full Faith and Credit Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
generalFundObligationLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: General Fund Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
revenueObligationLiability
An obligation and deliverable obligation characteristic. Defined in the ISDA published additional provisions for U.S. Municipal as Reference Entity. ISDA 2003 Term: Revenue Obligation Liability
Empty
type: Empty from: fpml-shared-4-8.xsd
notContingent
0..1
NOTE: Only allowed as an obligation charcteristic under ISDA Credit 1999. In essence Not Contingent means the repayment of principal cannot be dependant on a formula/index, i.e. to prevent the risk of being delivered an instrument that may never pay any element of principal, and to ensure that the obligation is interest bearing (on a regular schedule). ISDA 2003 Term: Not Contingent
Empty
type: Empty from: fpml-shared-4-8.xsd
excluded
0..1
A free format string to specify any excluded obligations or deliverable obligations, as the case may be, of the reference entity or excluded types of obligations or deliverable obligations. ISDA 2003 Term: Excluded Obligations/Excluded Deliverable Obligations
string
type: string from: XMLSchema.xsd
othReferenceEntityObligations
0..1
This element is used to specify any other obligations of a reference entity in both obligations and deliverable obligations. The obligations can be specified free-form. ISDA 2003 Term: Other Obligations of a Reference Entity



Schema Component Representation

```
<xsd:complexType name="Obligations">
  <xsd:sequence>
    <xsd:element name="category" type=" ObligationCategoryEnum "/>
    <xsd:element name="notSubordinated" type=" Empty " minOccurs="0"/>
    <xsd:element name="specifiedCurrency" type=" SpecifiedCurrency " minOccurs="0"/>
    <xsd:element name="notSovereignLender" type=" Empty " minOccurs="0"/>
    <xsd:element name="notDomesticCurrency" type=" NotDomesticCurrency " minOccurs="0"/>
    <xsd:element name="notDomesticLaw" type=" Empty " minOccurs="0"/>
    <xsd:element name="listed" type=" Empty " minOccurs="0"/>
    <xsd:element name="notDomesticIssuance" type=" Empty " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="fullFaithAndCreditObLiability" type=" Empty "/>
      <xsd:element name="generalFundObligationLiability" type=" Empty "/>
      <xsd:element name="revenueObligationLiability" type=" Empty "/>
    </xsd:choice>
    <xsd:element name="notContingent" type=" Empty " minOccurs="0"/>
    <xsd:element name="excluded" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="othReferenceEntityObligations" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="designatedPriority" type=" Lien " minOccurs="0"/>
    <xsd:element name="cashSettlementOnly" type=" Empty " minOccurs="0"/>
    <xsd:element name="deliveryOfCommitments" type=" Empty " minOccurs="0"/>
    <xsd:element name="continuity" type=" Empty " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PCDeliverableObligationCharac

[Table of contents]

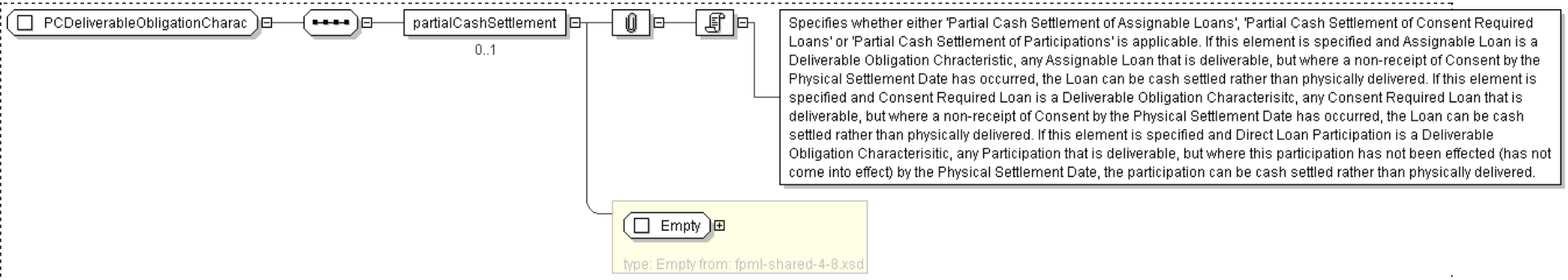
Super-types:	None
Sub-types:	• LoanParticipation (by extension)

Name	PCDeliverableObligationCharac
Used by (from the same schema document)	Complex Type DeliverableObligations , Complex Type DeliverableObligations
Abstract	no

XML Instance Representation

```
<...>
  <partialCashSettlement> Empty </partialCashSettlement> [0..1]
   'Specifies whether either \'Partial Cash Settlement of Assignable Loans\', \'Partial Cash Settlement of Consent Required Loans\' or \'Partial Cash
   Settlement of Participations\' is applicable. If this element is specified and Assignable Loan is a Deliverable Obligation Chracteristic, any
   Assignable Loan that is deliverable, but where a non-receipt of Consent by the Physical Settlement Date has occurred, the Loan can be cash settled
   rather than physically delivered. If this element is specified and Consent Required Loan is a Deliverable Obligation Characterisitc, any Consent
   Required Loan that is deliverable, but where a non-receipt of Consent by the Physical Settlement Date has occurred, the Loan can be cash settled
   rather than physically delivered. If this element is specified and Direct Loan Participation is a Deliverable Obligation Characterisitic, any
   Participation that is deliverable, but where this participation has not been effected (has not come into effect) by the Physical Settlement Date, the
   participation can be cash settled rather than physically delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PCDeliverableObligationCharac">
  <xsd:sequence>
    <xsd:element name="partialCashSettlement" type=" Empty " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PeriodicPayment

[Table of contents]

Super-types:	PaymentBase < PeriodicPayment (by extension)
Sub-types:	None

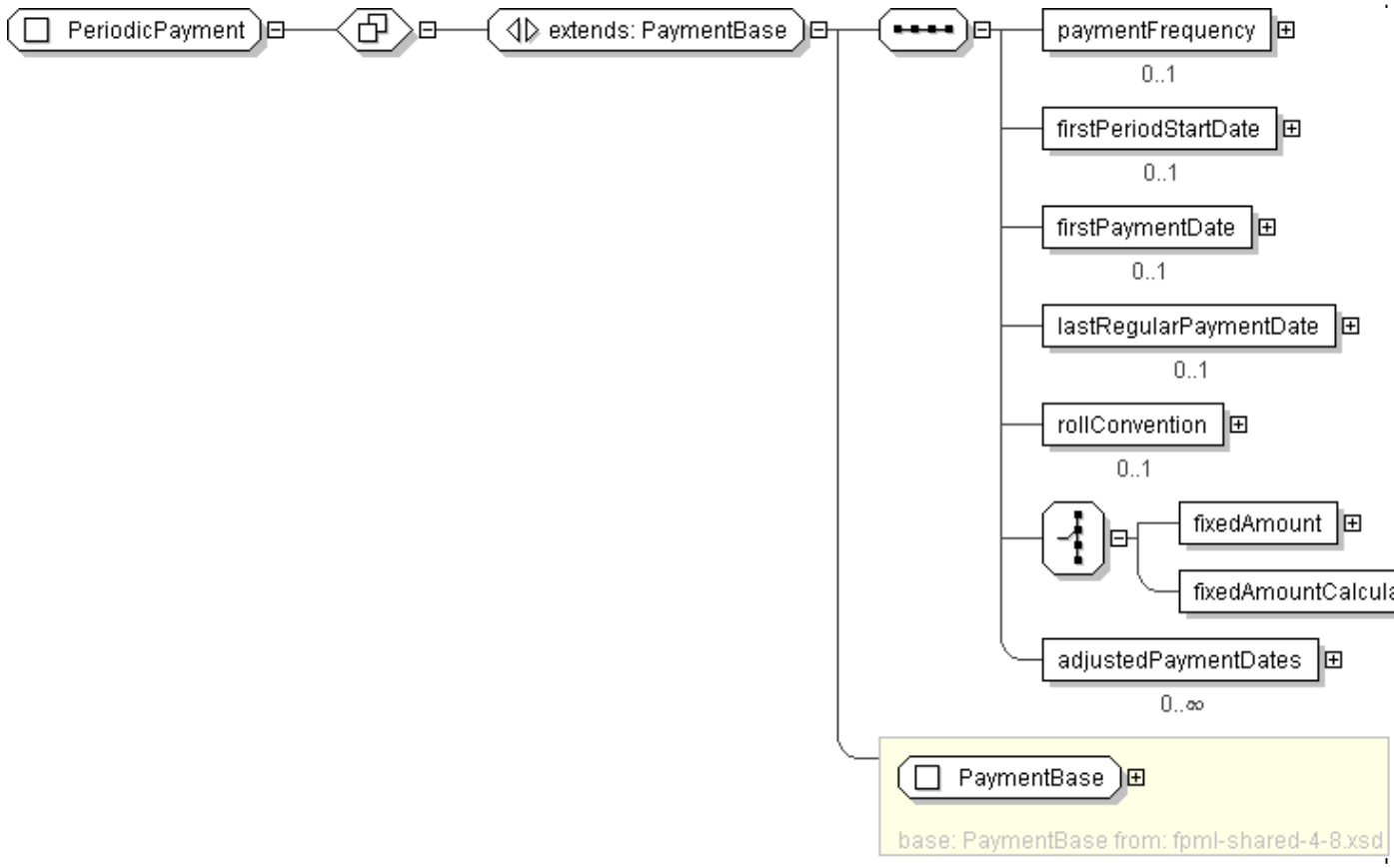
Name	PeriodicPayment
Used by (from the same schema document)	Complex Type FeeLeg
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <paymentFrequency> Period </paymentFrequency> [0..1]
   'The time interval between regular fixed rate payer payment dates.'

   <firstPeriodStartDate> xsd:date </firstPeriodStartDate> [0..1]
   'The start date of the initial calculation period if such date is not equal to the
   tradeâ€™s effective date. It must only be specified if it is not equal to the
   effective date. The applicable business day convention and business day are those
   specified in the dateAdjustments element within the generalTerms component (or in a
   transaction supplement FpML representation defined within the referenced general
   terms confirmation agreement).'

   <firstPaymentDate> xsd:date </firstPaymentDate> [0..1]
   'The first unadjusted fixed rate payer payment date. The applicable business day
   convention and business day are those specified in the dateAdjustments element
   within the generalTerms component (or in a transaction supplement FpML
   representation defined within the referenced general terms confirmation agreement).
   ISDA 2003 Term: Fixed Rate Payer Payment Date'

   <lastRegularPaymentDate> xsd:date </lastRegularPaymentDate> [0..1]
   'The last regular unadjusted fixed rate payer payment date. The applicable business
   day convention and business day are those specified in the dateAdjustments element
   within the generalTerms component (or in a transaction supplement FpML
   representation defined within the referenced general terms confirmation agreement).
   This element should only be included if there is a final payment stub, i.e. where
   the last regular unadjusted fixed rate payer payment date is not equal to the
   scheduled termination date. ISDA 2003 Term: Fixed Rate Payer Payment Date'

   <rollConvention> RollConventionEnum </rollConvention> [0..1]
   'Used in conjunction with the effectiveDate, scheduledTerminationDate,
   firstPaymentDate, lastRegularPaymentDate and paymentFrequency to determine the
   regular fixed rate payer payment dates.'

   Start Choice [1]
      <fixedAmount> Money </fixedAmount> [1]
      'A fixed payment amount. ISDA 2003 Term: Fixed Amount'

      <fixedAmountCalculation> FixedAmountCalculation </fixedAmountCalculation> [1]
      'This element contains all the terms relevant to calculating a fixed amount where
      the fixed amount is calculated by reference to a per annum fixed rate. There is
      no corresponding ISDA 2003 Term. The equivalent is Sec 5.1 \"Calculation of Fixed
      Amount\" but this in itself is not a defined Term.'

   End Choice
   <adjustedPaymentDates> AdjustedPaymentDates </adjustedPaymentDates> [0..*]
   'An optional cashflow-like structure allowing the equivalent representation of the
   periodic fixed payments in terms of a series of adjusted payment dates and amounts.
   This is intended to support application integration within an organisation and is
   not intended for use in inter-firm communication or confirmations. ISDA 2003 Term:
```

```
   Fixed Rate Payer Payment Date'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PeriodicPayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="paymentFrequency" type=" Period " minOccurs="0"/>
        <xsd:element name="firstPeriodStartDate" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="firstPaymentDate" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="lastRegularPaymentDate" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="rollConvention" type=" RollConventionEnum " minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="fixedAmount" type=" Money "/>
          <xsd:element name="fixedAmountCalculation" type=" FixedAmountCalculation "/>
        </xsd:choice>
        <xsd:element name="adjustedPaymentDates" type=" AdjustedPaymentDates "
        minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PhysicalSettlementPeriod

[Table of contents]

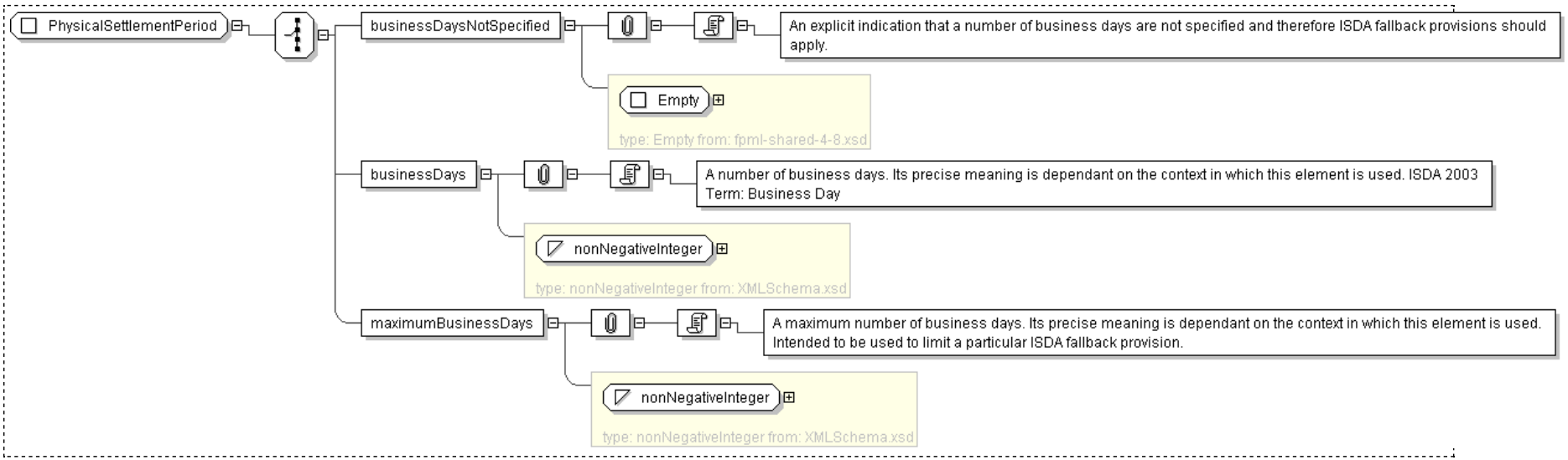
Super-types:	None
Sub-types:	None

Name	PhysicalSettlementPeriod
Used by (from the same schema document)	Complex Type PhysicalSettlementTerms
Abstract	no

XML Instance Representation

```
<...>
  Start Choice [1]
    <businessDaysNotSpecified> Empty </businessDaysNotSpecified> [1]
      'An explicit indication that a number of business days are not specified and therefore ISDA fallback provisions should apply.'

    <businessDays> xsd:nonNegativeInteger </businessDays> [1]
      'A number of business days. Its precise meaning is dependant on the context in which this element is used. ISDA 2003 Term: Business Day'

    <maximumBusinessDays> xsd:nonNegativeInteger </maximumBusinessDays> [1]
      'A maximum number of business days. Its precise meaning is dependant on the context in which this element is used. Intended to be used to
      limit a particular ISDA fallback provision.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PhysicalSettlementPeriod">
  <xsd:choice>
    <xsd:element name="businessDaysNotSpecified" type=" Empty "/>
    <xsd:element name="businessDays" type=" xsd:nonNegativeInteger "/>
    <xsd:element name="maximumBusinessDays" type=" xsd:nonNegativeInteger "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PhysicalSettlementTerms

[Table of contents]

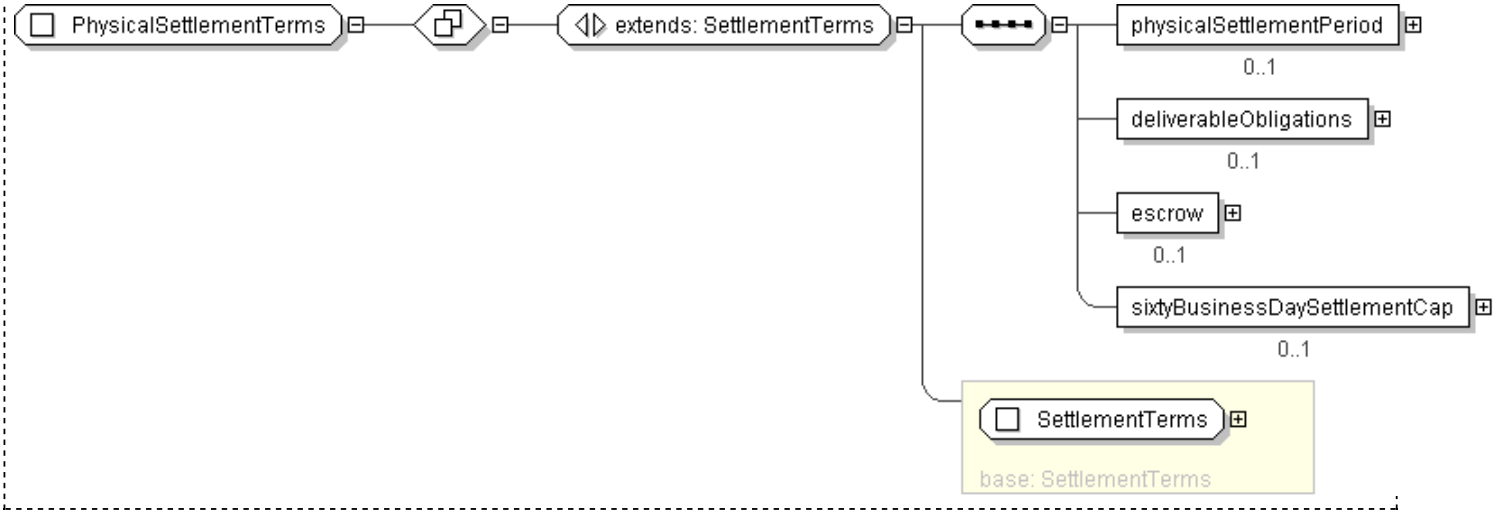
Super-types:	SettlementTerms < PhysicalSettlementTerms (by extension)
Sub-types:	None

Name	PhysicalSettlementTerms
Used by (from the same schema document)	Complex Type CreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <settlementCurrency> Currency </settlementCurrency> [0..1]
   'ISDA 2003 Term: Settlement Currency'

   <physicalSettlementPeriod> PhysicalSettlementPeriod </physicalSettlementPeriod> [0..1]
   'The number of business days used in the determination of the physical settlement date.
   The physical settlement date is this number of business days after all applicable
   conditions to settlement are satisfied. If a number of business days is not specified
   fallback provisions apply for determining the number of business days. If Section 8.5/8.6
   of the 1999/2003 ISDA Definitions are to apply the businessDaysNotSpecified element
   should be included. If a specified number of business days are to apply these should be
   specified in the businessDays element. If Section 8.5/8.6 of the 1999/2003 ISDA
   Definitions are to apply but capped at a maximum number of business days then the maximum
   number should be specified in the maximumBusinessDays element. ISDA 2003 Term: Physical
   Settlement Period'

   <deliverableObligations> DeliverableObligations </deliverableObligations> [0..1]
   'This element contains all the ISDA terms relevant to defining the deliverable
   obligations.'

   <escrow> xsd:boolean </escrow> [0..1]
   'If this element is specified, indicates that physical settlement must take place through
   the use of an escrow agent. (For Canadian counterparties this is always \"Not
   Applicable\". ISDA 2003 Term: Escrow'

   <sixtyBusinessDaySettlementCap> xsd:boolean </sixtyBusinessDaySettlementCap> [0..1]
   'If this element is specified, for a transaction documented under the 2003 ISDA Credit
   Derivatives Definitions, has the effect of incorporating the language set forth below
   into the confirmation. The section references are to the 2003 ISDA Credit Derivatives
   Definitions. Notwithstanding Section 1.7 or any provisions of Sections 9.9 or 9.10 to the
   contrary, but without prejudice to Section 9.3 and (where applicable) Sections 9.4, 9.5
   and 9.6, if the Termination Date has not occurred on or prior to the date that is 60
   Business Days following the Physical Settlement Date, such 60th Business Day shall be
   deemed to be the Termination Date with respect to this Transaction except in relation to
   any portion of the Transaction (an \"Affected Portion\") in respect of which: (1) a valid
   notice of Buy-in Price has been delivered that is effective fewer than three Business
   Days prior to such 60th Business Day, in which case the Termination Date for that
   Affected Portion shall be the third Business Day following the date on which such notice
   is effective; or (2) Buyer has purchased but not Delivered Deliverable Obligations
   validly specified by Seller pursuant to Section 9.10(b), in which case the Termination
   Date for that Affected Portion shall be the tenth Business Day following the date on
   which Seller validly specified such Deliverable Obligations to Buyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PhysicalSettlementTerms">
  <xsd:complexContent>
    <xsd:extension base=" SettlementTerms ">
      <xsd:sequence>
        <xsd:element name="physicalSettlementPeriod" type=" PhysicalSettlementPeriod "
        minOccurs="0"/>
        <xsd:element name="deliverableObligations" type=" DeliverableObligations "
        minOccurs="0"/>
        <xsd:element name="escrow" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="sixtyBusinessDaySettlementCap" type=" xsd:boolean " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProtectionTerms

[Table of contents]

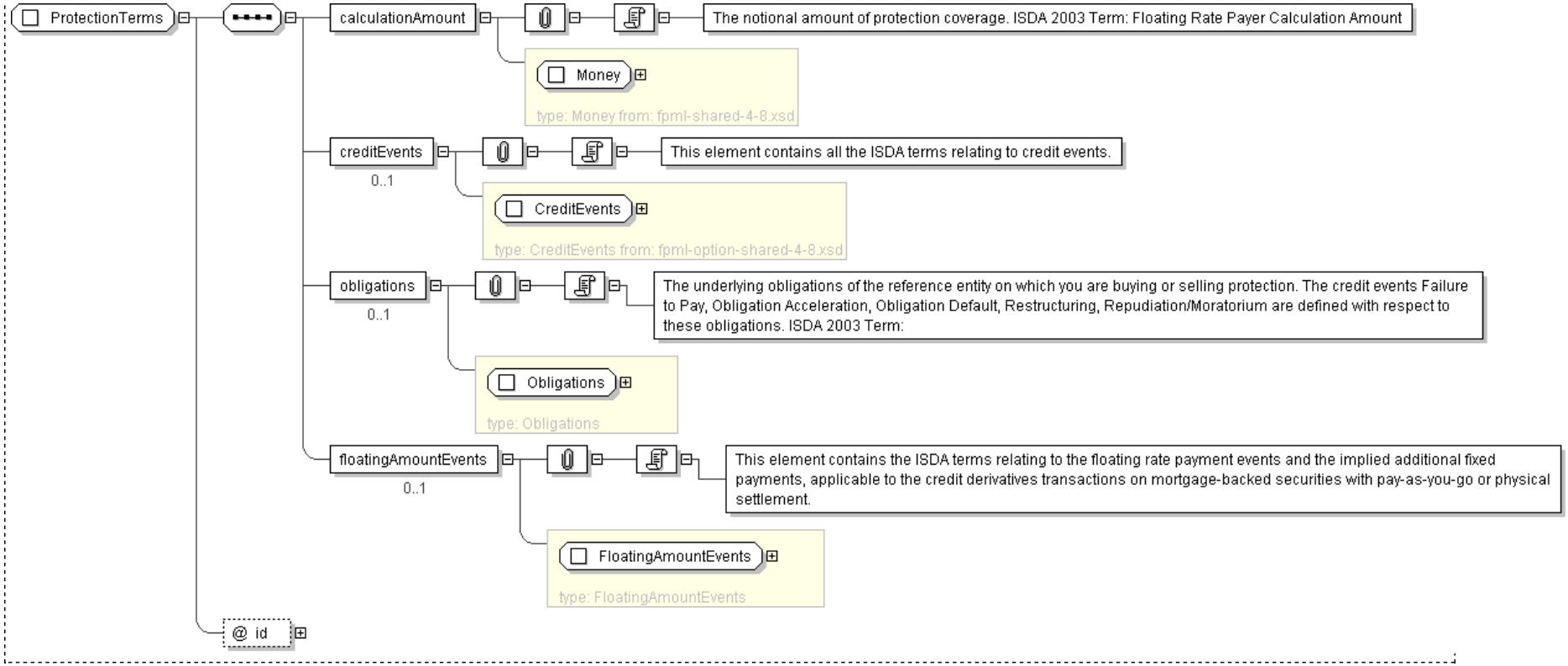
Super-types:	None
Sub-types:	None

Name	ProtectionTerms
Used by (from the same schema document)	Complex Type CreditDefaultSwap
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <calculationAmount> Money </calculationAmount> [1]
   'The notional amount of protection coverage. ISDA 2003 Term: Floating Rate Payer Calculation Amount'

   <creditEvents> CreditEvents </creditEvents> [0..1]
   'This element contains all the ISDA terms relating to credit events.'

   <obligations> Obligations </obligations> [0..1]
   'The underlying obligations of the reference entity on which you are buying or selling protection. The credit events Failure to Pay,
   Obligation Acceleration, Obligation Default, Restructuring, Repudiation/Moratorium are defined with respect to these obligations. ISDA 2003
   Term:'

   <floatingAmountEvents> FloatingAmountEvents </floatingAmountEvents> [0..1]
   'This element contains the ISDA terms relating to the floating rate payment events and the implied additional fixed payments, applicable to
   the credit derivatives transactions on mortgage-backed securities with pay-as-you-go or physical settlement.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProtectionTerms">
  <xsd:sequence>
    <xsd:element name="calculationAmount" type=" Money "/>
    <xsd:element name="creditEvents" type=" CreditEvents " minOccurs="0"/>
    <xsd:element name="obligations" type=" Obligations " minOccurs="0"/>
    <xsd:element name="floatingAmountEvents" type=" FloatingAmountEvents " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProtectionTermsReference

[Table of contents]

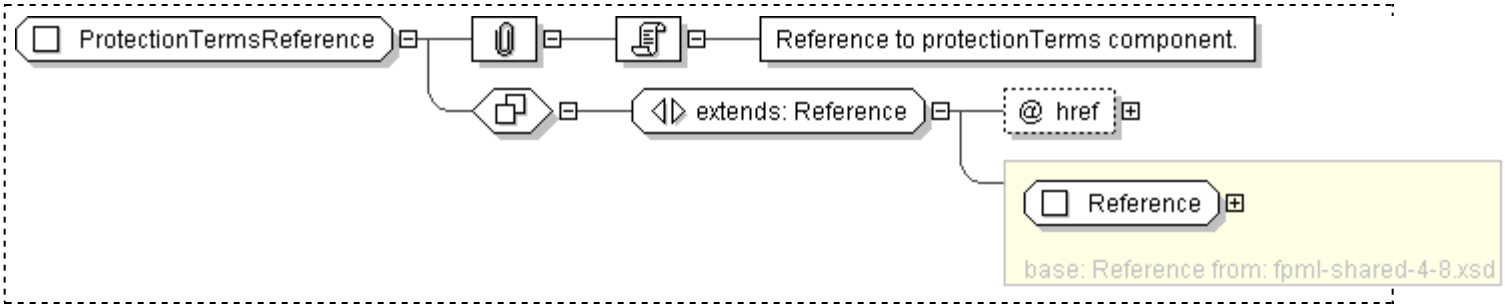
Super-types:	Reference < ProtectionTermsReference (by extension)
Sub-types:	None

Name	ProtectionTermsReference
Used by (from the same schema document)	Complex Type ReferencePoolItem
Abstract	no
Documentation	Reference to protectionTerms component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProtectionTermsReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="ProtectionTerms"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceInformation

[Table of contents]

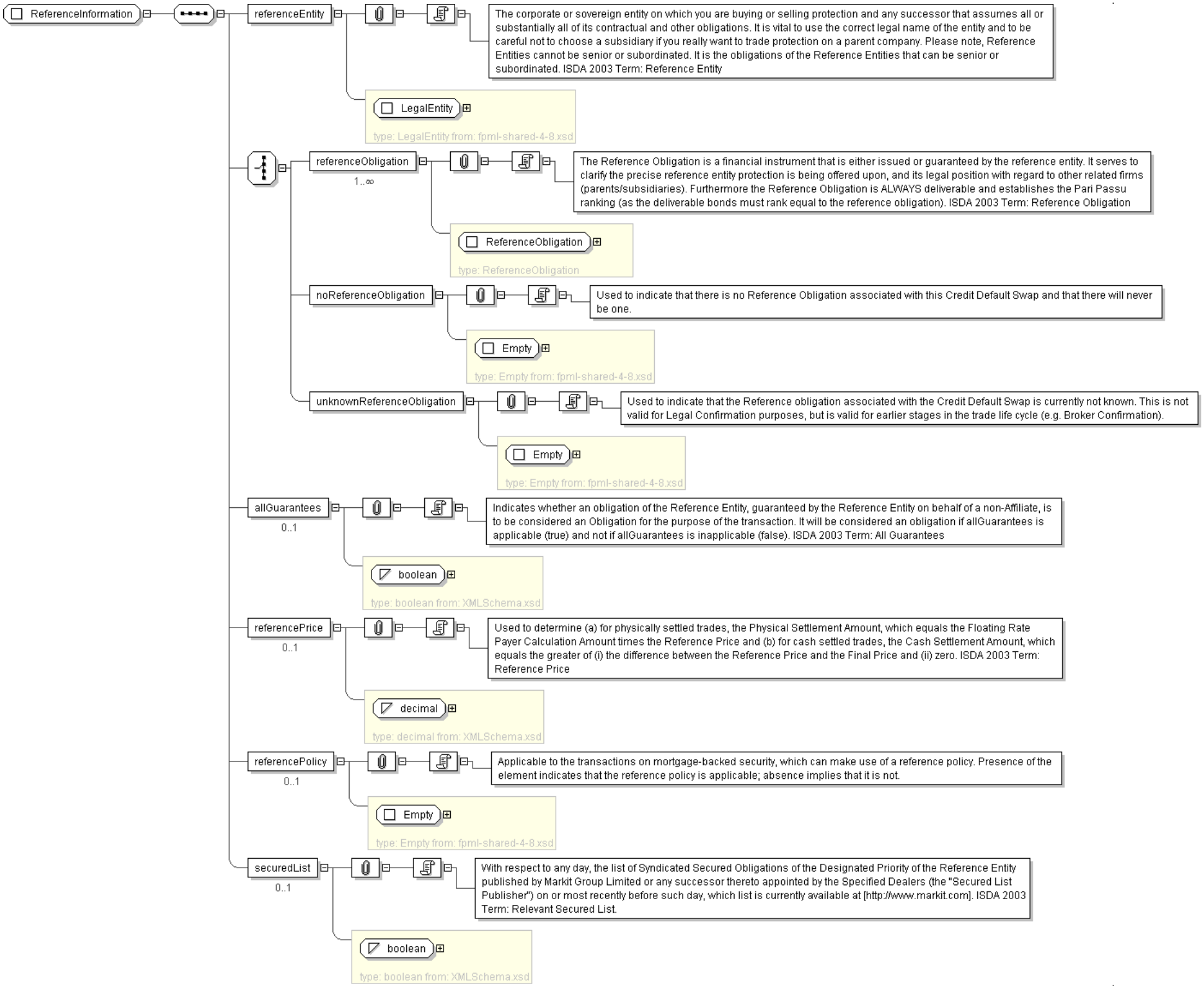
Super-types:	None
Sub-types:	None

Name	ReferenceInformation
Used by (from the same schema document)	Complex Type GeneralTerms
Abstract	no

XML Instance Representation

```
<...>
   <referenceEntity> LegalEntity </referenceEntity> [1]
   'The corporate or sovereign entity on which you are buying or selling protection and any successor that assumes all or substantially all of its contractual
   and other obligations. It is vital to use the correct legal name of the entity and to be careful not to choose a subsidiary if you really want to trade
   protection on a parent company. Please note, Reference Entities cannot be senior or subordinated. It is the obligations of the Reference Entities that can
   be senior or subordinated. ISDA 2003 Term: Reference Entity'

   Start Choice [1]
      <referenceObligation> ReferenceObligation </referenceObligation> [1..*]
      'The Reference Obligation is a financial instrument that is either issued or guaranteed by the reference entity. It serves to clarify the precise
      reference entity protection is being offered upon, and its legal position with regard to other related firms (parents/subsidiaries). Furthermore the
      Reference Obligation is ALWAYS deliverable and establishes the Pari Passu ranking (as the deliverable bonds must rank equal to the reference obligation).
      ISDA 2003 Term: Reference Obligation'

      <noReferenceObligation> Empty </noReferenceObligation> [1]
      'Used to indicate that there is no Reference Obligation associated with this Credit Default Swap and that there will never be one.'

      <unknownReferenceObligation> Empty </unknownReferenceObligation> [1]
      'Used to indicate that the Reference obligation associated with the Credit Default Swap is currently not known. This is not valid for Legal Confirmation
      purposes, but is valid for earlier stages in the trade life cycle (e.g. Broker Confirmation).'

   End Choice
   <allGuarantees> xsd:boolean </allGuarantees> [0..1]
   'Indicates whether an obligation of the Reference Entity, guaranteed by the Reference Entity on behalf of a non-Affiliate, is to be considered an Obligation
   for the purpose of the transaction. It will be considered an obligation if allGuarantees is applicable (true) and not if allGuarantees is inapplicable
   (false). ISDA 2003 Term: All Guarantees'

   <referencePrice> xsd:decimal </referencePrice> [0..1]
   'Used to determine (a) for physically settled trades, the Physical Settlement Amount, which equals the Floating Rate Payer Calculation Amount times the
   Reference Price and (b) for cash settled trades, the Cash Settlement Amount, which equals the greater of (i) the difference between the Reference Price and
   the Final Price and (ii) zero. ISDA 2003 Term: Reference Price'

   <referencePolicy> Empty </referencePolicy> [0..1]
   'Applicable to the transactions on mortgage-backed security, which can make use of a reference policy. Presence of the element indicates that the reference
   policy is applicable; absence implies that it is not.'

   <securedList> xsd:boolean </securedList> [0..1]
   'With respect to any day, the list of Syndicated Secured Obligations of the Designated Priority of the Reference Entity published by Markit Group Limited or
   any successor thereto appointed by the Specified Dealers (the \"Secured List Publisher\") on or most recently before such day, which list is currently
   available at [http://www.markit.com]. ISDA 2003 Term: Relevant Secured List.'

</...>
```

Diagram


ReferenceInformation
referenceEntity
The corporate or sovereign entity on which you are buying or selling protection and any successor that assumes all or substantially all of its contractual and other obligations. It is vital to use the correct legal name of the entity and to be careful not to choose a subsidiary if you really want to trade protection on a parent company. Please note, Reference Entities cannot be senior or subordinated. It is the obligations of the Reference Entities that can be senior or subordinated. ISDA 2003 Term: Reference Entity
LegalEntity
type: LegalEntity from: fpml-shared-4-8.xsd
referenceObligation
1..∞
The Reference Obligation is a financial instrument that is either issued or guaranteed by the reference entity. It serves to clarify the precise reference entity protection is being offered upon, and its legal position with regard to other related firms (parents/subsidiaries). Furthermore the Reference Obligation is ALWAYS deliverable and establishes the Pari Passu ranking (as the deliverable bonds must rank equal to the reference obligation). ISDA 2003 Term: Reference Obligation
ReferenceObligation
type: ReferenceObligation
noReferenceObligation
Used to indicate that there is no Reference Obligation associated with this Credit Default Swap and that there will never be one.
Empty
type: Empty from: fpml-shared-4-8.xsd
unknownReferenceObligation
Used to indicate that the Reference obligation associated with the Credit Default Swap is currently not known. This is not valid for Legal Confirmation purposes, but is valid for earlier stages in the trade life cycle (e.g. Broker Confirmation).
Empty
type: Empty from: fpml-shared-4-8.xsd
allGuarantees
0..1
Indicates whether an obligation of the Reference Entity, guaranteed by the Reference Entity on behalf of a non-Affiliate, is to be considered an Obligation for the purpose of the transaction. It will be considered an obligation if allGuarantees is applicable (true) and not if allGuarantees is inapplicable (false). ISDA 2003 Term: All Guarantees
boolean
type: boolean from: XMLSchema.xsd
referencePrice
0..1
Used to determine (a) for physically settled trades, the Physical Settlement Amount, which equals the Floating Rate Payer Calculation Amount times the Reference Price and (b) for cash settled trades, the Cash Settlement Amount, which equals the greater of (i) the difference between the Reference Price and the Final Price and (ii) zero. ISDA 2003 Term: Reference Price
decimal
type: decimal from: XMLSchema.xsd
referencePolicy
0..1
Applicable to the transactions on mortgage-backed security, which can make use of a reference policy. Presence of the element indicates that the reference policy is applicable; absence implies that it is not.
Empty
type: Empty from: fpml-shared-4-8.xsd
securedList
0..1
With respect to any day, the list of Syndicated Secured Obligations of the Designated Priority of the Reference Entity published by Markit Group Limited or any successor thereto appointed by the Specified Dealers (the "Secured List Publisher") on or most recently before such day, which list is currently available at [http://www.markit.com]. ISDA 2003 Term: Relevant Secured List.
boolean
type: boolean from: XMLSchema.xsd

Schema Component Representation

```
<xsd:complexType name="ReferenceInformation">
   <xsd:sequence>
      <xsd:element name="referenceEntity" type=" LegalEntity "/>
      <xsd:choice>
         <xsd:element name="referenceObligation" type=" ReferenceObligation " maxOccurs="unbounded"/>
         <xsd:element name="noReferenceObligation" type=" Empty "/>
         <xsd:element name="unknownReferenceObligation" type=" Empty "/>
      </xsd:choice>
      <xsd:element name="allGuarantees" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="referencePrice" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="referencePolicy" type=" Empty " minOccurs="0"/>
      <xsd:element name="securedList" type=" xsd:boolean " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceObligation

[Table of contents]

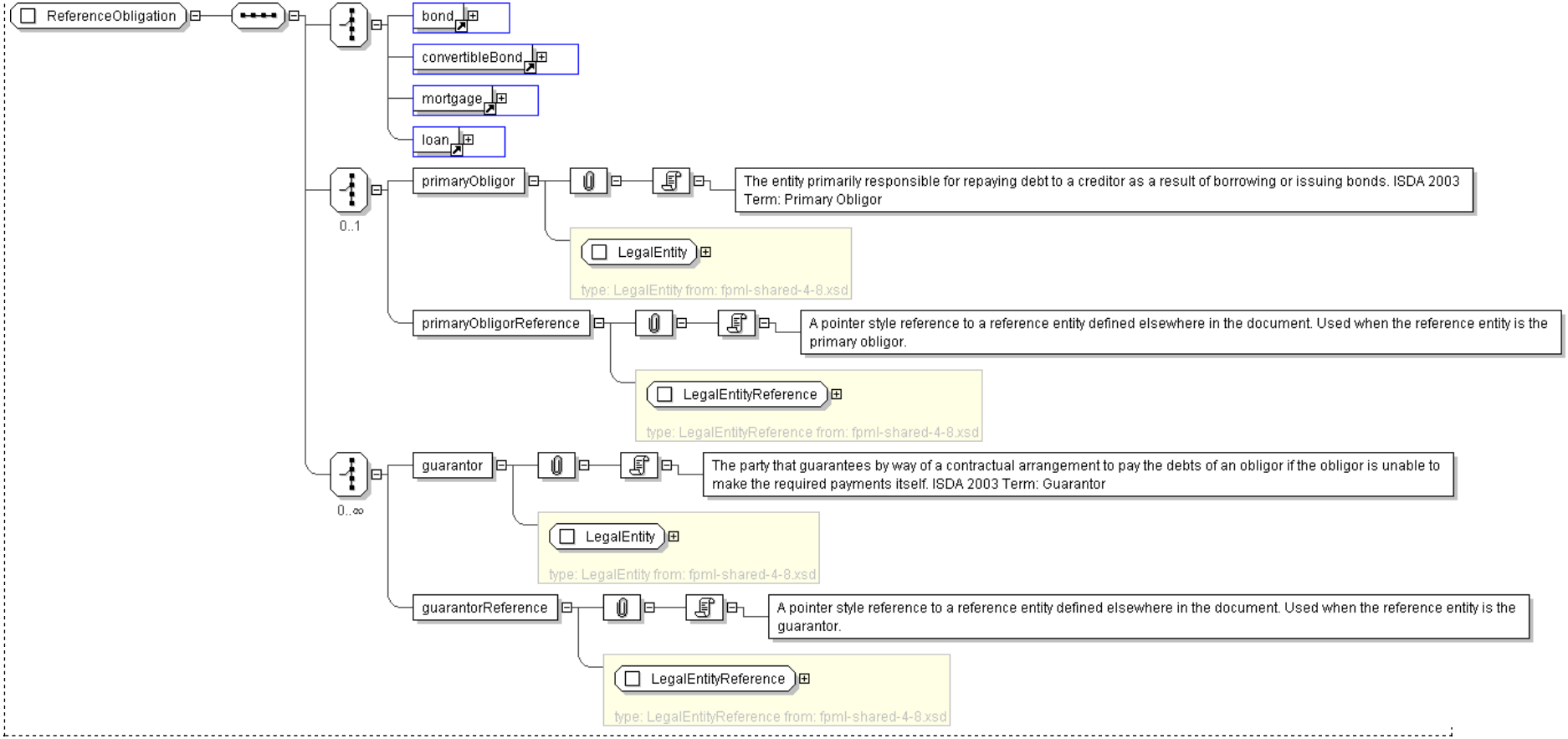
Super-types:	None
Sub-types:	None

Name	ReferenceObligation
Used by (from the same schema document)	Complex Type ReferenceInformation , Complex Type ReferencePair
Abstract	no

XML Instance Representation

```
<...>
   Start Choice [1]
      <bond> ... </bond> [1]
      <convertibleBond> ... </convertibleBond> [1]
      <mortgage> ... </mortgage> [1]
      <loan> ... </loan> [1]
   End Choice
   Start Choice [0..1]
      <primaryObligor> LegalEntity </primaryObligor> [1]
      'The entity primarily responsible for repaying debt to a creditor as a result of borrowing or issuing bonds. ISDA 2003 Term: Primary Obligor'

      <primaryObligorReference> LegalEntityReference </primaryObligorReference> [1]
      'A pointer style reference to a reference entity defined elsewhere in the document. Used when the reference entity is the primary obligor.'

   End Choice
   Start Choice [0..*]
      <guarantor> LegalEntity </guarantor> [1]
      'The party that guarantees by way of a contractual arrangement to pay the debts of an obligor if the obligor is unable to make the required payments
      itself. ISDA 2003 Term: Guarantor'

      <guarantorReference> LegalEntityReference </guarantorReference> [1]
      'A pointer style reference to a reference entity defined elsewhere in the document. Used when the reference entity is the guarantor.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferenceObligation">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element ref=" bond "/>
      <xsd:element ref=" convertibleBond "/>
      <xsd:element ref=" mortgage "/>
      <xsd:element ref=" loan "/>
    </xsd:choice>
    <xsd:choice minOccurs="0">
      <xsd:element name="primaryObligor" type=" LegalEntity "/>
      <xsd:element name="primaryObligorReference" type=" LegalEntityReference "/>
    </xsd:choice>
    <xsd:choice minOccurs="0" maxOccurs="unbounded">
      <xsd:element name="guarantor" type=" LegalEntity "/>
      <xsd:element name="guarantorReference" type=" LegalEntityReference "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferencePair

[Table of contents]

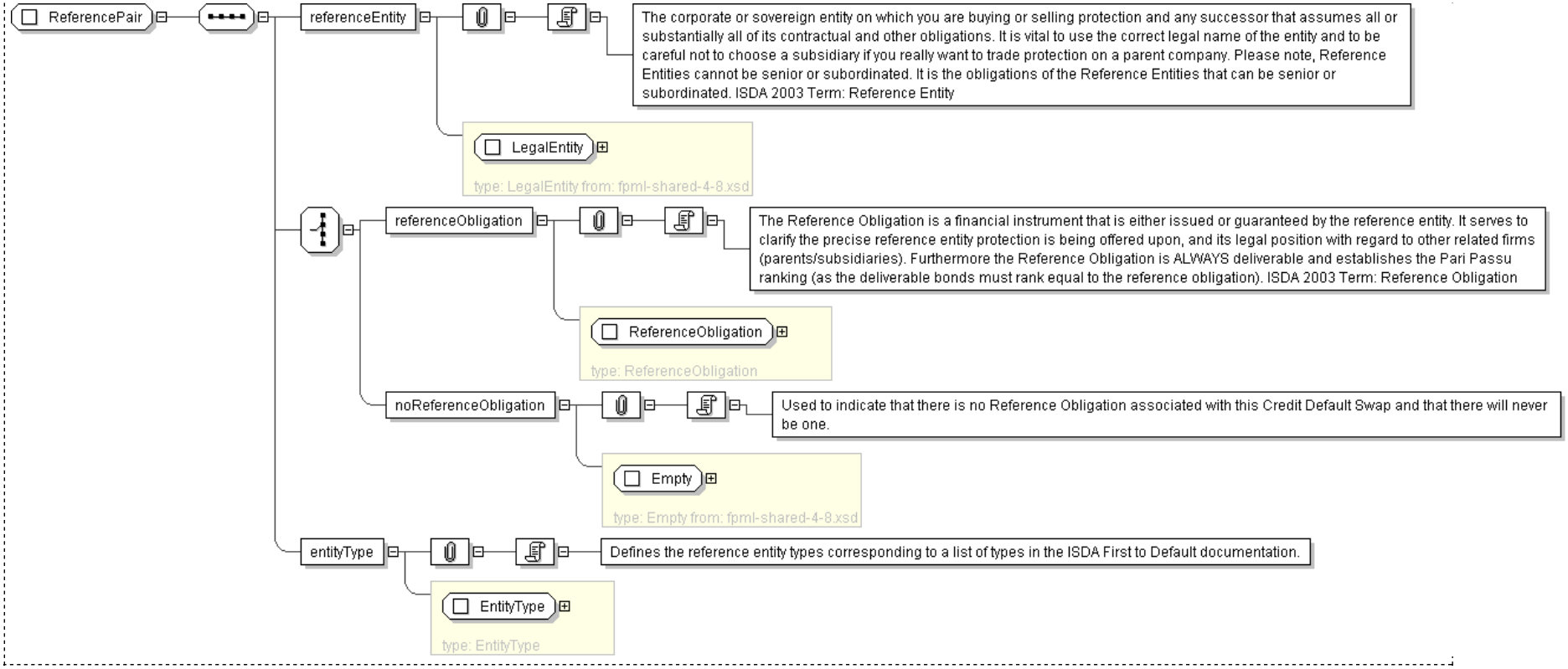
Super-types:	None
Sub-types:	None

Name	ReferencePair
Used by (from the same schema document)	Complex Type ReferencePoolItem
Abstract	no

XML Instance Representation

```
<...>
   <referenceEntity> LegalEntity </referenceEntity> [1]
   'The corporate or sovereign entity on which you are buying or selling protection and any successor that assumes all or substantially all of its
   contractual and other obligations. It is vital to use the correct legal name of the entity and to be careful not to choose a subsidiary if you
   really want to trade protection on a parent company. Please note, Reference Entities cannot be senior or subordinated. It is the obligations of the
   Reference Entities that can be senior or subordinated. ISDA 2003 Term: Reference Entity'

   Start Choice [1]
      <referenceObligation> ReferenceObligation </referenceObligation> [1]
      'The Reference Obligation is a financial instrument that is either issued or guaranteed by the reference entity. It serves to clarify the
      precise reference entity protection is being offered upon, and its legal position with regard to other related firms (parents/subsidiaries).
      Furthermore the Reference Obligation is ALWAYS deliverable and establishes the Pari Passu ranking (as the deliverable bonds must rank equal to
      the reference obligation). ISDA 2003 Term: Reference Obligation'

      <noReferenceObligation> Empty </noReferenceObligation> [1]
      'Used to indicate that there is no Reference Obligation associated with this Credit Default Swap and that there will never be one.'

   End Choice
   <entityType> EntityType </entityType> [1]
   'Defines the reference entity types corresponding to a list of types in the ISDA First to Default documentation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferencePair">
  <xsd:sequence>
    <xsd:element name="referenceEntity" type=" LegalEntity "/>
    <xsd:choice>
      <xsd:element name="referenceObligation" type=" ReferenceObligation "/>
      <xsd:element name="noReferenceObligation" type=" Empty "/>
    </xsd:choice>
    <xsd:element name="entityType" type=" EntityType "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferencePool

[Table of contents]

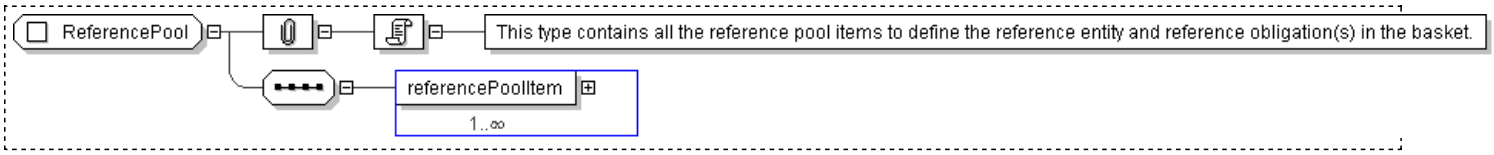
Super-types:	None
Sub-types:	None

Name	ReferencePool
Used by (from the same schema document)	Complex Type BasketReferenceInformation
Abstract	no
Documentation	This type contains all the reference pool items to define the reference entity and reference obligation(s) in the basket.

XML Instance Representation

```
<...>
  <referencePoolItem> ReferencePoolItem </referencePoolItem> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferencePool">
  <xsd:sequence>
    <xsd:element name="referencePoolItem" type=" ReferencePoolItem " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferencePoolItem

[Table of contents]

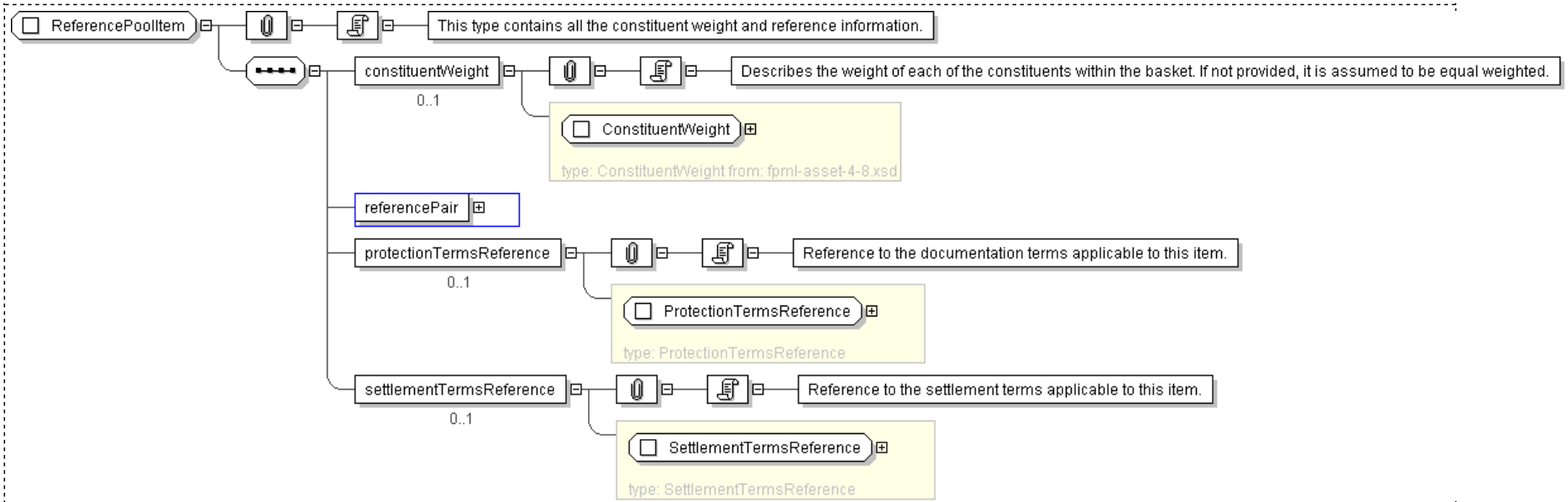
Super-types:	None
Sub-types:	None

Name	ReferencePoolItem
Used by (from the same schema document)	Complex Type ReferencePool
Abstract	no
Documentation	This type contains all the constituent weight and reference information.

XML Instance Representation

```
<...>
   <constituentWeight> ConstituentWeight </constituentWeight> [0..1]
   'Describes the weight of each of the constituents within the basket. If not provided, it is assumed to be equal weighted.'

   <referencePair> ReferencePair </referencePair> [1]
   <protectionTermsReference> ProtectionTermsReference </protectionTermsReference> [0..1]
   'Reference to the documentation terms applicable to this item.'

   <settlementTermsReference> SettlementTermsReference </settlementTermsReference> [0..1]
   'Reference to the settlement terms applicable to this item.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferencePoolItem">
  <xsd:sequence>
    <xsd:element name="                 " type="                 " minOccurs=" "/>
```

```
                        constituentWeight         ConstituentWeight              0
      <xsd:element name="referencePair" type=" ReferencePair "/>
      <xsd:element name="protectionTermsReference" type=" ProtectionTermsReference " minOccurs="0"/>
      <xsd:element name="settlementTermsReference" type=" SettlementTermsReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ScheduledTerminationDate

[Table of contents]

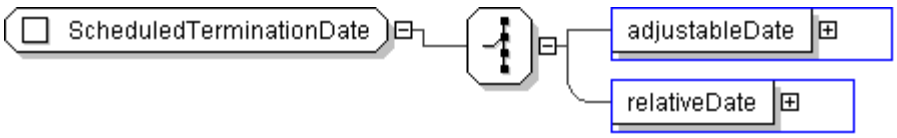
Super-types:	None
Sub-types:	None

Name	ScheduledTerminationDate
Abstract	no

XML Instance Representation

```
<...>
   Start Choice [1]
      <adjustableDate> AdjustableDate2 </adjustableDate> [1]
      <relativeDate> Period </relativeDate> [1]
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ScheduledTerminationDate">
   <xsd:choice>
      <xsd:element name="adjustableDate" type=" AdjustableDate2 "/>
      <xsd:element name="relativeDate" type=" Period "/>
   </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettledEntityMatrix

[Table of contents]

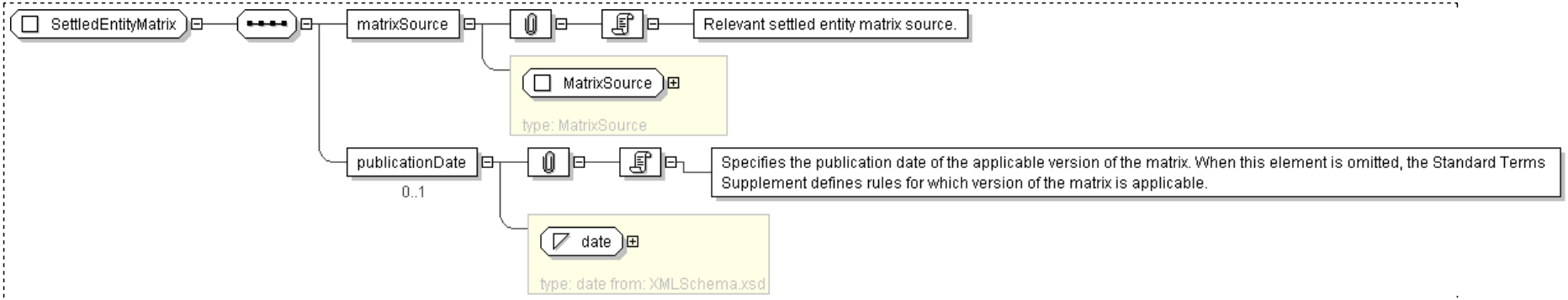
Super-types:	None
Sub-types:	None

Name	SettledEntityMatrix
Used by (from the same schema document)	Complex Type IndexReferenceInformation
Abstract	no

XML Instance Representation

```
<...>
  <matrixSource> MatrixSource </matrixSource> [1]
  'Relevant settled entity matrix source.'

  <publicationDate> xsd:date </publicationDate> [0..1]
  'Specifies the publication date of the applicable version of the matrix. When this element is omitted, the Standard Terms Supplement
  defines rules for which version of the matrix is applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettledEntityMatrix">
  <xsd:sequence>
    <xsd:element name="matrixSource" type=" MatrixSource "/>
    <xsd:element name="publicationDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementTerms

[Table of contents]

Super-types: None

Sub-types:

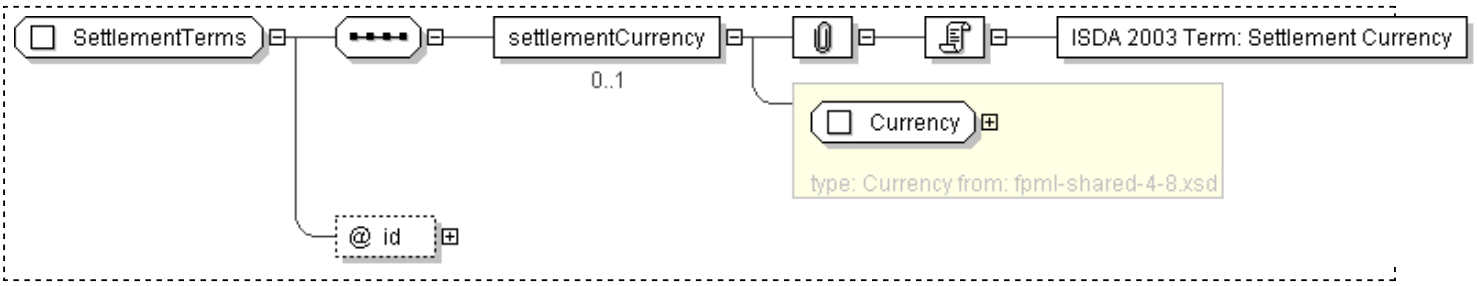
- CashSettlementTerms (by extension)
- PhysicalSettlementTerms (by extension)

Name	SettlementTerms
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <settlementCurrency> Currency </settlementCurrency> [0..1]
   'ISDA 2003 Term: Settlement Currency'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementTerms">
   <xsd:sequence>
      <xsd:element name="settlementCurrency" type=" Currency " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementTermsReference

[Table of contents]

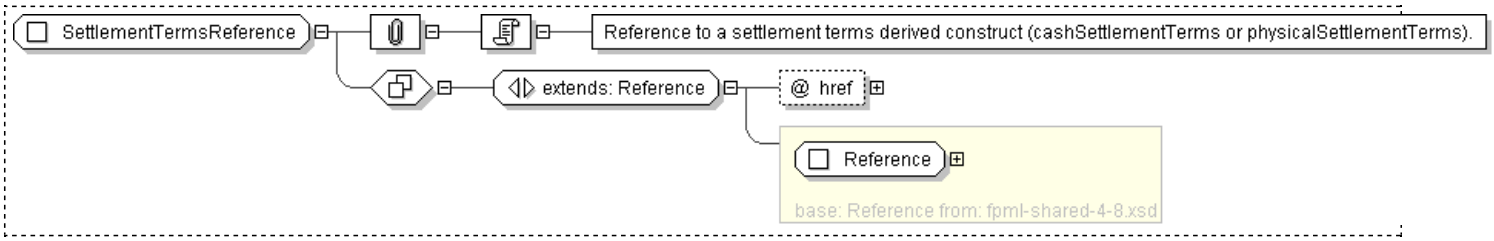
Super-types:	Reference < SettlementTermsReference (by extension)
Sub-types:	None

Name	SettlementTermsReference
Used by (from the same schema document)	Complex Type ReferencePoolItem
Abstract	no
Documentation	Reference to a settlement terms derived construct (cashSettlementTerms or physicalSettlementTerms).

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementTermsReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="SettlementTerms"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SinglePayment

[Table of contents]

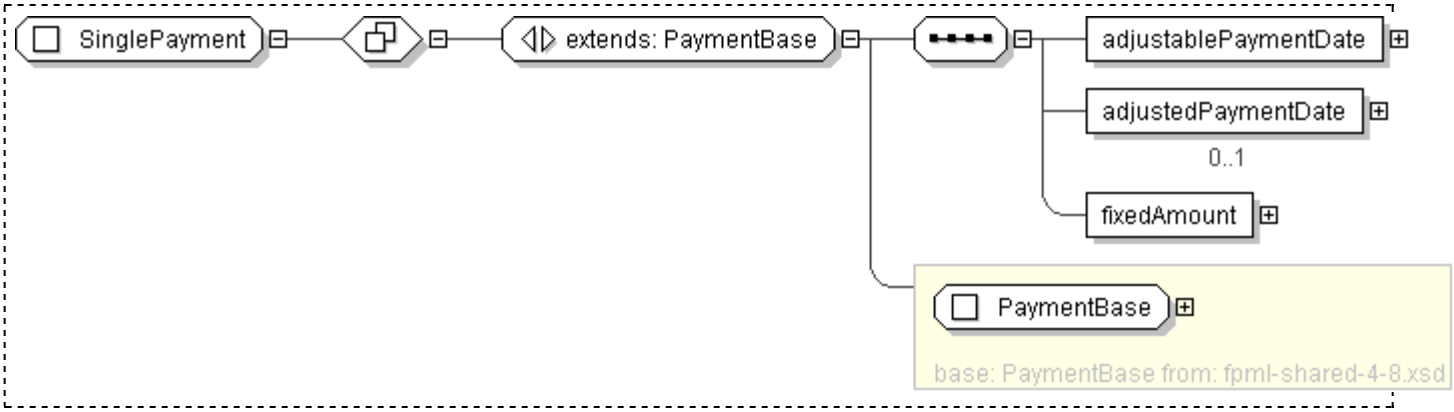
Super-types:	PaymentBase < SinglePayment (by extension)
Sub-types:	None

Name	SinglePayment
Used by (from the same schema document)	Complex Type FeeLeg
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <adjustablePaymentDate> xsd:date </adjustablePaymentDate> [1]
    'A fixed amount payment date that shall be subject to adjustment in accordance
    with the applicable business day convention if it would otherwise fall on a day
    that is not a business day. The applicable business day convention and business
    day are those specified in the dateAdjustments element within the generalTerms
    component. ISDA 2003 Term: Fixed Rate Payer Payment Date'

   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [0..1]
    'The adjusted payment date. This date should already be adjusted for any
    applicable business day convention. This component is not intended for use in
    trade confirmation but may be specified to allow the fee structure to also
    serve as a cashflow type component.'

   <fixedAmount> Money </fixedAmount> [1]
    'A fixed payment amount. ISDA 2003 Term: Fixed Amount'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SinglePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:element name="adjustablePaymentDate" type=" xsd:date "/>
            <xsd:element name="adjustedPaymentDate" type=" xsd:date " minOccurs="0"/>
            <xsd:element name="fixedAmount" type=" Money "/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SingleValuationDate

[Table of contents]

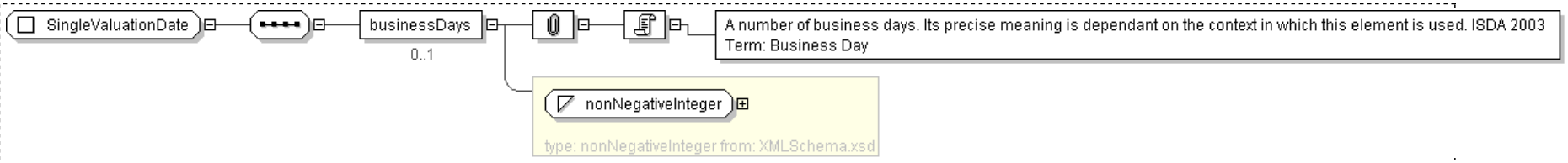
Super-types:	None
Sub-types:	• MultipleValuationDates (by extension)

Name	SingleValuationDate
Used by (from the same schema document)	Complex Type ValuationDate
Abstract	no

XML Instance Representation

```
<...>
  <businessDays> xsd:nonNegativeInteger </businessDays> [0..1]
  'A number of business days. Its precise meaning is dependant on the context in which this element is used. ISDA 2003 Term: Business Day'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SingleValuationDate">
  <xsd:sequence>
    <xsd:element name="businessDays" type=" xsd:nonNegativeInteger " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SpecifiedCurrency

[Table of contents]

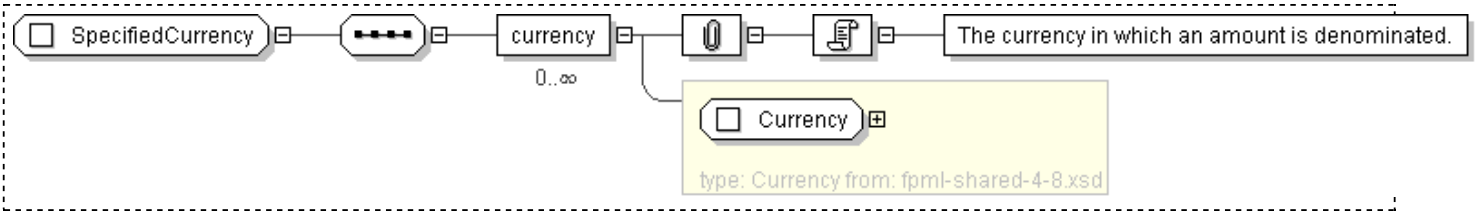
Super-types:	None
Sub-types:	None

Name	SpecifiedCurrency
Used by (from the same schema document)	Complex Type DeliverableObligations , Complex Type Obligations
Abstract	no

XML Instance Representation

```
<...>
   <currency> Currency </currency> [0..*]
   'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SpecifiedCurrency">
   <xsd:sequence>
      <xsd:element name="currency" type=" Currency " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Tranche

[Table of contents]

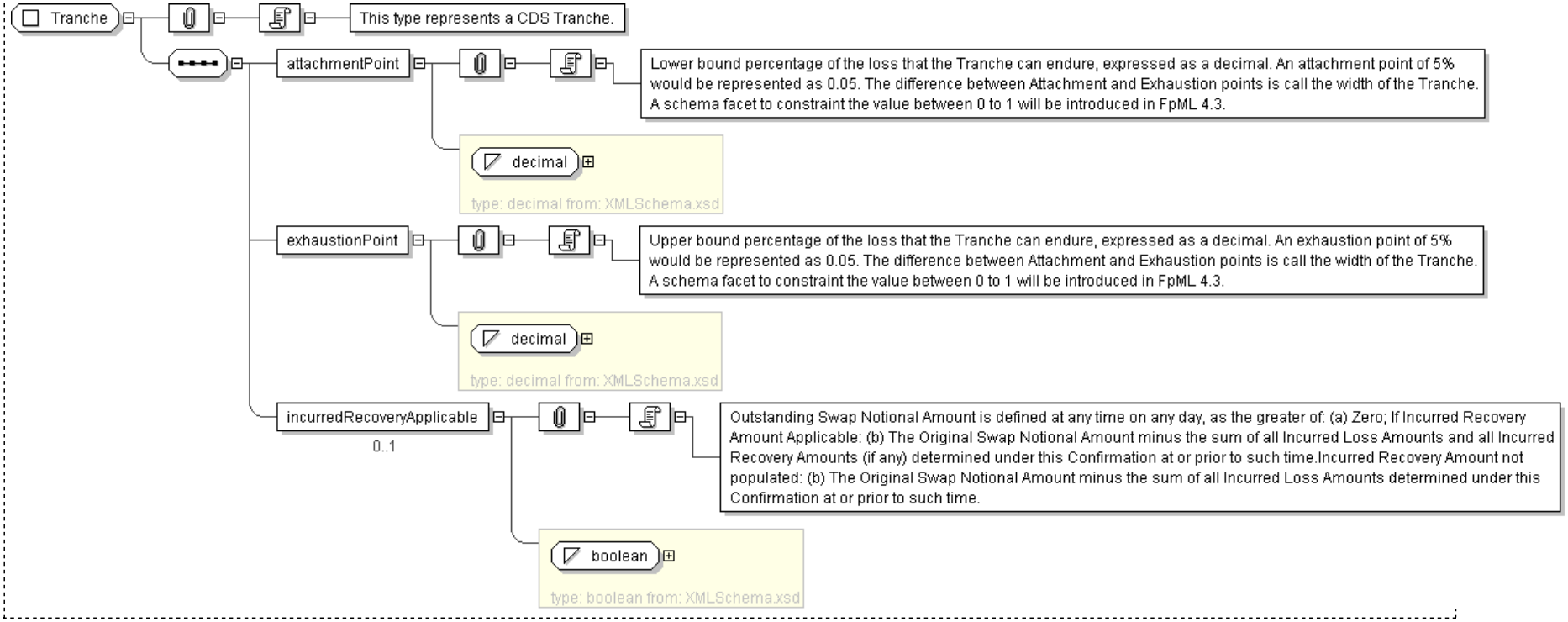
Super-types:	None
Sub-types:	None

Name	Tranche
Used by (from the same schema document)	Complex Type BasketReferenceInformation , Complex Type IndexReferenceInformation
Abstract	no
Documentation	This type represents a CDS Tranche.

XML Instance Representation

```
<...>
   <attachmentPoint> xsd:decimal </attachmentPoint> [1]
   'Lower bound percentage of the loss that the Tranche can endure, expressed as a decimal. An attachment point of 5% would be represented
   as 0.05. The difference between Attachment and Exhaustion points is call the width of the Tranche. A schema facet to constraint the value
   between 0 to 1 will be introduced in FpML 4.3.'

   <exhaustionPoint> xsd:decimal </exhaustionPoint> [1]
   'Upper bound percentage of the loss that the Tranche can endure, expressed as a decimal. An exhaustion point of 5% would be represented
   as 0.05. The difference between Attachment and Exhaustion points is call the width of the Tranche. A schema facet to constraint the value
   between 0 to 1 will be introduced in FpML 4.3.'

   <incurredRecoveryApplicable> xsd:boolean </incurredRecoveryApplicable> [0..1]
   'Outstanding Swap Notional Amount is defined at any time on any day, as the greater of: (a) Zero; If Incurred Recovery Amount Applicable:
   (b) The Original Swap Notional Amount minus the sum of all Incurred Loss Amounts and all Incurred Recovery Amounts (if any) determined
   under this Confirmation at or prior to such time.Incurred Recovery Amount not populated: (b) The Original Swap Notional Amount minus the
   sum of all Incurred Loss Amounts determined under this Confirmation at or prior to such time.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Tranche">
  <xsd:sequence>
    <xsd:element name="attachmentPoint" type=" xsd:decimal "/>
    <xsd:element name="exhaustionPoint" type=" xsd:decimal "/>
    <xsd:element name="incurredRecoveryApplicable" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationDate

[Table of contents]

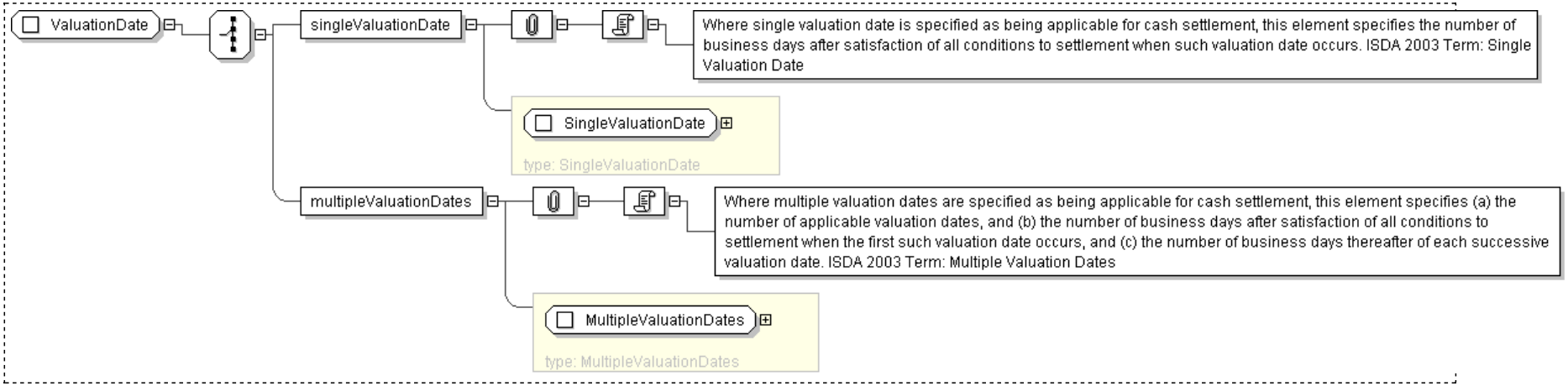
Super-types:	None
Sub-types:	None

Name	ValuationDate
Used by (from the same schema document)	Complex Type CashSettlementTerms
Abstract	no

XML Instance Representation

```
<...>
  Start Choice [1]
    <singleValuationDate> SingleValuationDate </singleValuationDate> [1]
     'Where single valuation date is specified as being applicable for cash settlement, this element specifies the number of business days
     after satisfaction of all conditions to settlement when such valuation date occurs. ISDA 2003 Term: Single Valuation Date'

    <multipleValuationDates> MultipleValuationDates </multipleValuationDates> [1]
     'Where multiple valuation dates are specified as being applicable for cash settlement, this element specifies (a) the number of
     applicable valuation dates, and (b) the number of business days after satisfaction of all conditions to settlement when the first such
     valuation date occurs, and (c) the number of business days thereafter of each successive valuation date. ISDA 2003 Term: Multiple
     Valuation Dates'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationDate">
  <xsd:choice>
    <xsd:element name="singleValuationDate" type=" SingleValuationDate "/>
    <xsd:element name="multipleValuationDates" type=" MultipleValuationDates "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: additionalCommodityForwardLeg
  - Element: additionalCommoditySwapLeg
  - Element: commodityForward
  - Element: commodityOption
  - Element: commoditySwap
- Global Definitions
  - Complex Type: AbsoluteTolerance
  - Complex Type: BullionDeliveryLocation
  - Complex Type: BullionPhysicalLeg
  - Complex Type: CalculationPeriodsDatesReference
  - Complex Type: CalculationPeriodsReference
  - Complex Type: CalculationPeriodsScheduleReference
  - Complex Type: CoalAttributeDecimal
  - Complex Type: CoalAttributePercentage
  - Complex Type: CoalDelivery
  - Complex Type: CoalDeliveryPoint
  - Complex Type: CoalPhysicalLeg
  - Complex Type: CoalProduct
  - Complex Type: CoalProductSource
  - Complex Type: CoalProductSpecifications
  - Complex Type: CoalProductType
  - Complex Type: CoalQualityAdjustments
  - Complex Type: CoalStandardQuality
  - Complex Type: CoalStandardQualitySchedule
  - Complex Type: CoalTransportationEquipment
  - Complex Type: CommodityAmericanExercise
  - Complex Type: CommodityCalculationPeriodsSchedule
  - Complex Type: CommodityDeliveryPeriods
  - Complex Type: CommodityDeliveryPoint
  - Complex Type: CommodityDeliveryRisk
  - Complex Type: CommodityEuropeanExercise
  - Complex Type: CommodityExercise
  - Complex Type: CommodityExpireRelativeToEvent
  - Complex Type: CommodityFixedPriceSchedule
  - Complex Type: CommodityForward
  - Complex Type: CommodityFrequencyType
  - Complex Type: CommodityFx
  - Complex Type: CommodityFxType
  - Complex Type: CommodityHub
  - Complex Type: CommodityHubCode
  - Complex Type: CommodityMarketDisruption
  - Complex Type: CommodityMultipleExercise
  - Complex Type: CommodityNotionalQuantity
  - Complex Type: CommodityNotionalQuantitySchedule
  - Complex Type: CommodityOption
  - Complex Type: CommodityPayRelativeToEvent
  - Complex Type: CommodityPhysicalAmericanExercise
  - Complex Type: CommodityPhysicalEuropeanExercise
  - Complex Type: CommodityPhysicalExercise
  - Complex Type: CommodityPhysicalQuantity
  - Complex Type: CommodityPhysicalQuantityBase
  - Complex Type: CommodityPhysicalQuantitySchedule

- Complex Type: CommodityPipeline
- Complex Type: CommodityPipelineCycle
- Complex Type: CommodityPremium
- Complex Type: CommodityPricingDates
- Complex Type: CommodityProductGrade
- Complex Type: CommodityQuantityFrequency
- Complex Type: CommodityRelativeExpirationDates
- Complex Type: CommodityRelativePaymentDates
- Complex Type: CommoditySettlementPeriodsNotionalQuantity
- Complex Type: CommoditySettlementPeriodsNotionalQuantitySchedule
- Complex Type: CommoditySettlementPeriodsPriceSchedule
- Complex Type: CommoditySpreadSchedule
- Complex Type: CommodityStrikeSchedule
- Complex Type: CommoditySwap
- Complex Type: DisruptionFallback
- Complex Type: ElectricityDelivery
- Complex Type: ElectricityDeliveryFirm
- Complex Type: ElectricityDeliveryPeriods
- Complex Type: ElectricityDeliveryPoint
- Complex Type: ElectricityDeliverySystemFirm
- Complex Type: ElectricityDeliveryUnitFirm
- Complex Type: ElectricityPhysicalDeliveryQuantity
- Complex Type: ElectricityPhysicalDeliveryQuantitySchedule
- Complex Type: ElectricityPhysicalLeg
- Complex Type: ElectricityPhysicalQuantity
- Complex Type: ElectricityProduct
- Complex Type: ElectricityTransmissionContingency
- Complex Type: ElectricityTransmissionContingencyType
- Complex Type: FixedPrice
- Complex Type: FixedPriceLeg
- Complex Type: FloatingLegCalculation
- Complex Type: FloatingPriceLeg
- Complex Type: GasDelivery
- Complex Type: GasDeliveryPeriods
- Complex Type: GasDeliveryPoint
- Complex Type: GasPhysicalLeg
- Complex Type: GasPhysicalQuantity
- Complex Type: GasProduct
- Complex Type: GasQuality
- Complex Type: Lag
- Complex Type: LagReference
- Complex Type: MarketDisruptionEvent
- Complex Type: NonPeriodicFixedPriceLeg
- Complex Type: OilDelivery
- Complex Type: OilPhysicalLeg
- Complex Type: OilPipelineDelivery
- Complex Type: OilProduct
- Complex Type: OilProductType
- Complex Type: OilTransferDelivery
- Complex Type: PercentageTolerance
- Complex Type: PhysicalLeg
- Complex Type: QuantityReference
- Complex Type: QuantityScheduleReference
- Complex Type: SequencedDisruptionFallback
- Complex Type: SettlementPeriods
- Complex Type: SettlementPeriodsFixedPrice
- Complex Type: SettlementPeriodsReference
- Complex Type: SettlementPeriodsSchedule
- Complex Type: SettlementPeriodsStep
- Complex Type: UnitQuantity

- Model Group: CommodityAsian.model
- Model Group: CommodityCalculationPeriods.model
- Model Group: CommodityCalculationPeriodsPointer.model
- Model Group: CommodityCoalComposition.model
- Model Group: CommodityCoalProperties.model
- Model Group: CommodityCoalReducingAtmosphere.model
- Model Group: CommodityContent.model
- Model Group: CommodityDeliveryPeriodsPointer.model
- Model Group: CommodityDeliveryPoints.model
- Model Group: CommodityFinancialOption.model
- Model Group: CommodityFixedPhysicalQuantity.model
- Model Group: CommodityFixedPrice.model
- Model Group: CommodityFreightFlatRate.model
- Model Group: CommodityNonPeriodicPaymentDates.model
- Model Group: CommodityNotionalQuantity.model
- Model Group: CommodityPaymentDates.model
- Model Group: CommodityPhysicalOption.model
- Model Group: CommodityStrikePrice.model
- Model Group: CommodityUSCoalDelivery.model
- Model Group: CommodityUSCoalProduct.model
- Model Group: LagOrReference.model
- Model Group: Price.model
- Model Group: PricingDays.model

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 2406 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-option-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
```

```
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 2406 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-option-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
```

```
                <restriction base=" string ">
                   <pattern value="[1-9][0-9]{3}"/>
                </restriction>
             </simpleType>
          </element>
       </sequence>
       <attribute name="country" type=" string " fixed="Australia"/>
     </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

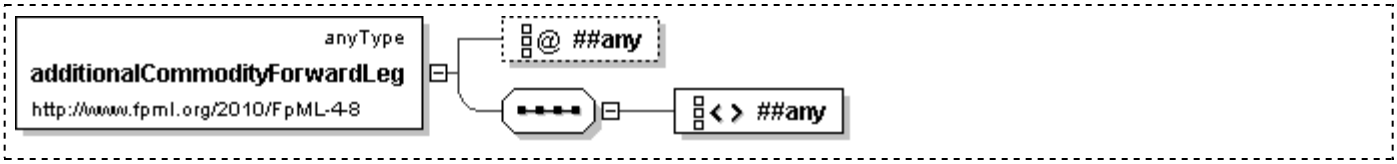
XML Schema Documentation

Element: additionalCommodityForwardLeg

[Table of contents]

Name	additionalCommodityForwardLeg
Used by (from the same schema document)	Complex Type CommodityForward
Type	anyType
Nillable	no
Abstract	yes

Logical Diagram



XML Instance Representation

```
<additionalCommodityForwardLeg> ... </additionalCommodityForwardLeg>
```

Diagram

Schema Component Representation

```
<xsd:element name="additionalCommodityForwardLeg" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

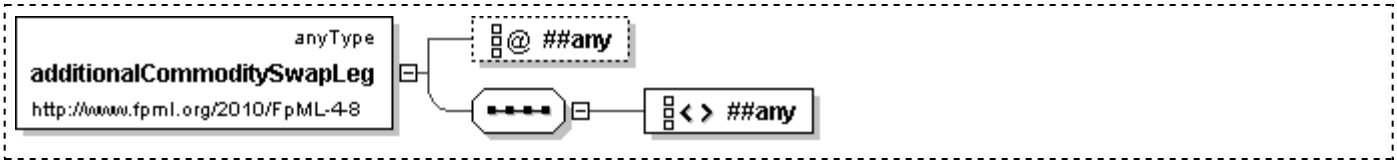
XML Schema Documentation

Element: additionalCommoditySwapLeg

[Table of contents]

Name	additionalCommoditySwapLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Type	anyType
Nillable	no
Abstract	yes

Logical Diagram



XML Instance Representation

```
<additionalCommoditySwapLeg> ... </additionalCommoditySwapLeg>
```

Diagram

Schema Component Representation

```
<xsd:element name="additionalCommoditySwapLeg" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

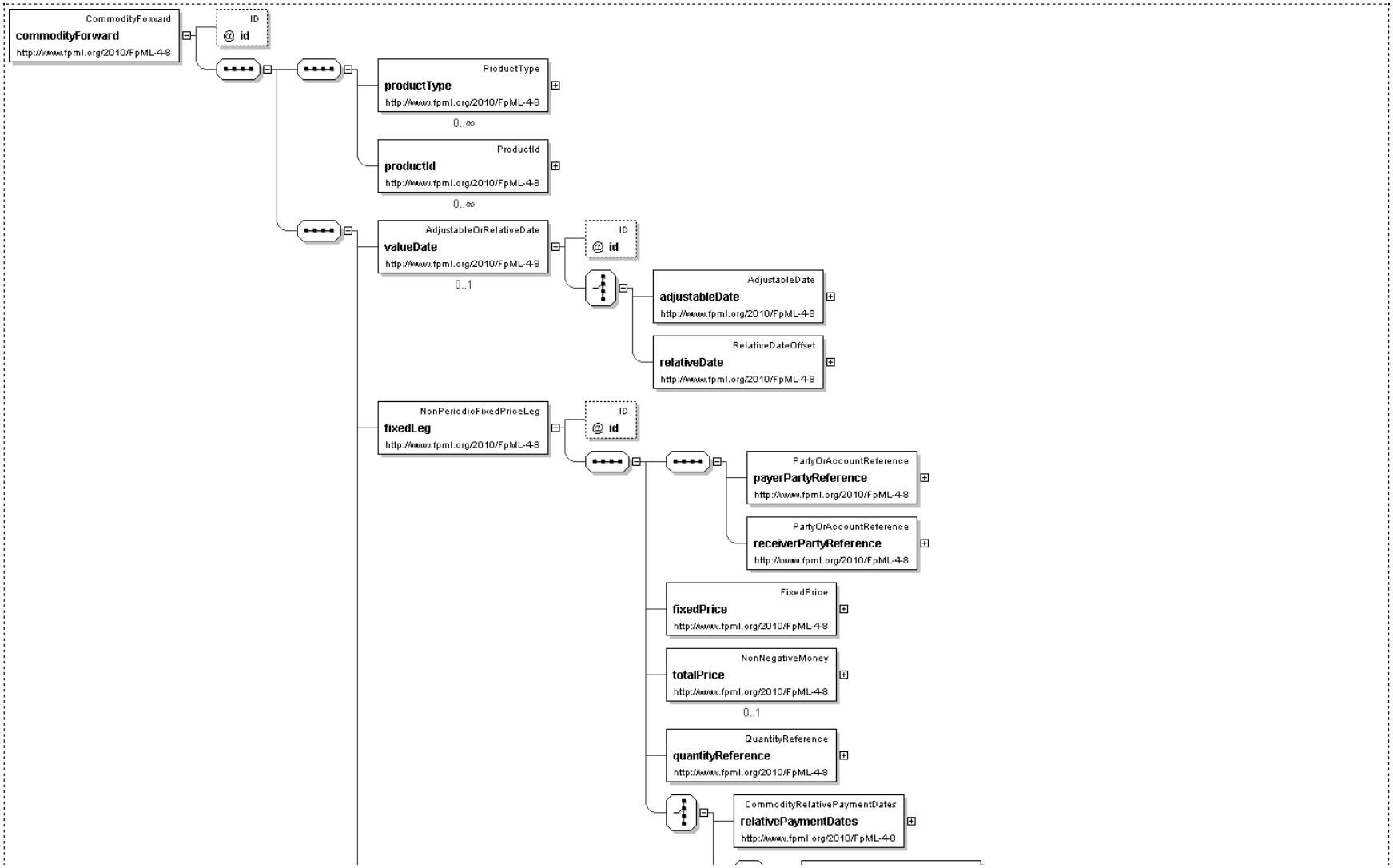
Element: commodityForward

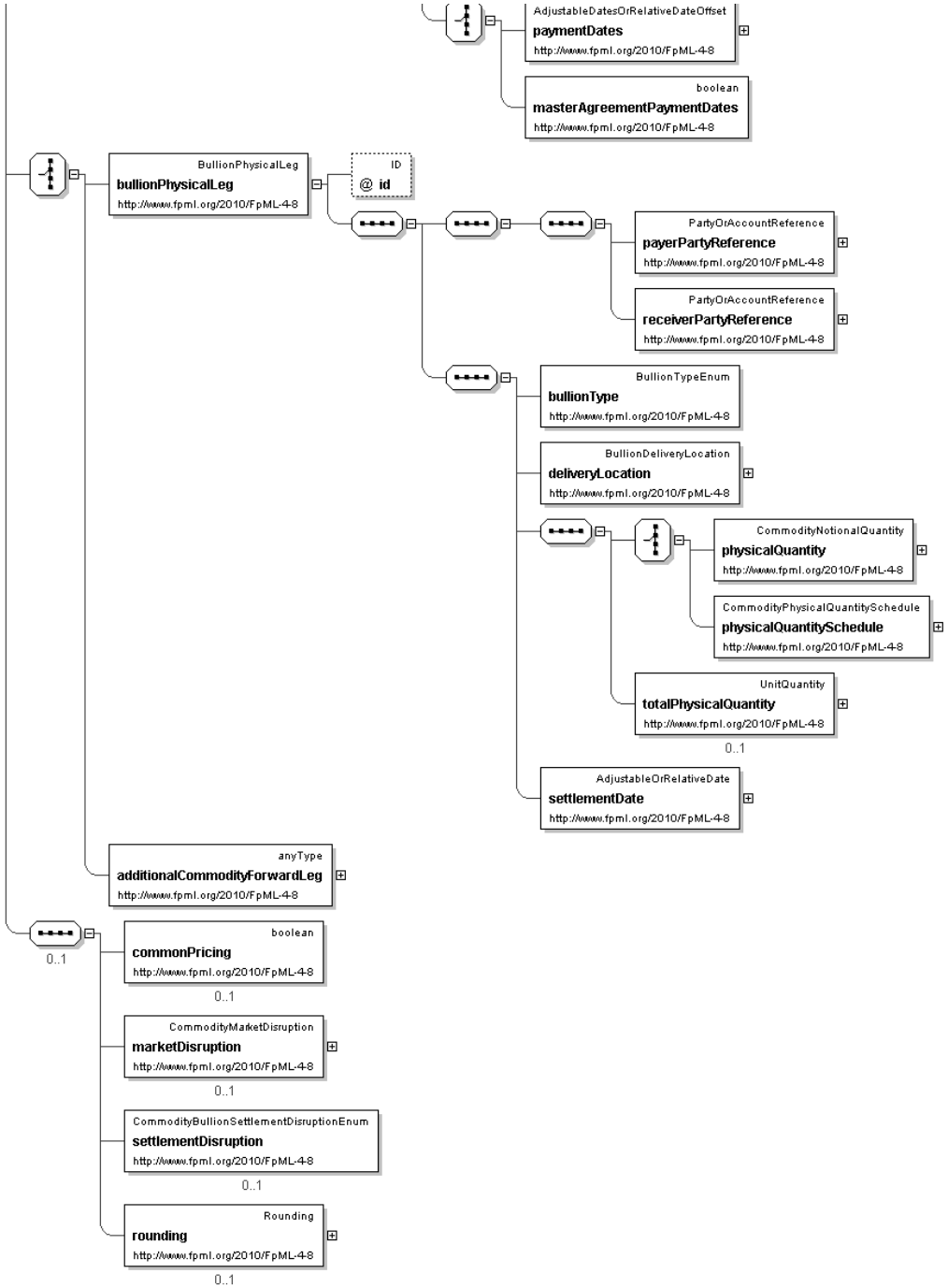
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	commodityForward
Used by (from the same schema document)	Model Group CommodityPhysicalOption.model
Type	CommodityForward
Nillable	no
Abstract	no
Documentation	Defines a commodity forward product.

Logical Diagram



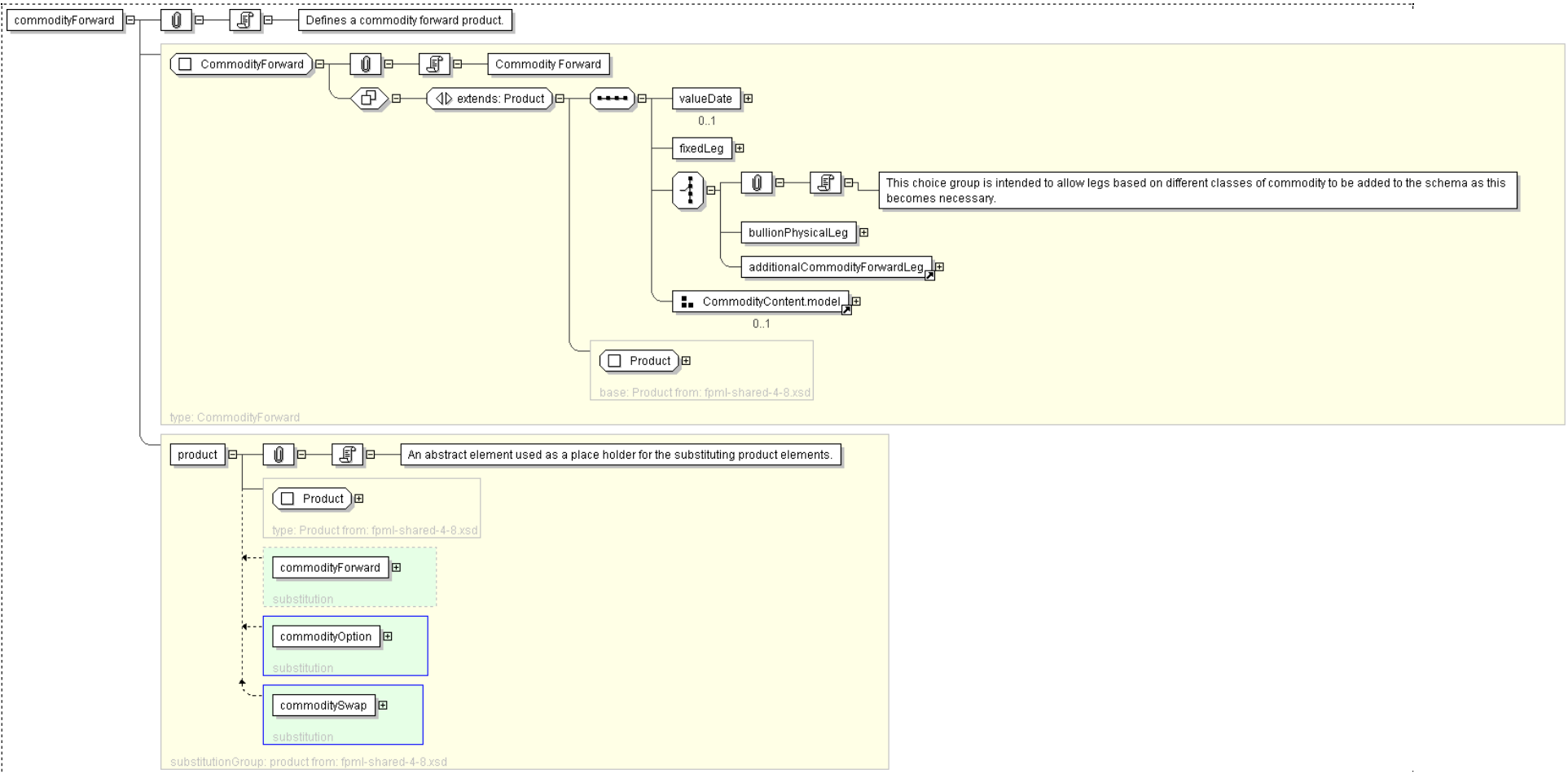


XML Instance Representation

```
<commodityForward
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not
```

```
   strictly an enumerated list.'

   <valueDate> AdjustableOrRelativeDate </valueDate> [0..1]
   'Specifies the value date of the Commodity Forward Transaction. This is the day on which both the cash and the physical commodity settle.'

   <fixedLeg> NonPeriodicFixedPriceLeg </fixedLeg> [1]
   'The fixed leg of a Commodity Forward Transaction'

   Start Choice [1]
'This choice group is intended to allow legs based on different classes of commodity to be added to the schema as this becomes necessary.'


      <bullionPhysicalLeg> BullionPhysicalLeg </bullionPhysicalLeg> [1]
      'The physical leg of a Commodity Forward Transaction for which the underlyer is Bullion.'

      <additionalCommodityForwardLeg> ... </additionalCommodityForwardLeg> [1]
   End Choice
   Start Group: CommodityContent.model [0..1]
      <commonPricing> xsd:boolean </commonPricing> [0..1]
      'Common pricing may be relevant for a Transaction that references more than one Commodity Reference Price. If Common Pricing is not specified as applicable, it will be deemed
      not to apply.'

      <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
      'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA 2005 Commodity Definitions, as applicable.'

      <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption> [0..1]
      'The consequences of Bullion Settlement Disruption Events.'

      <rounding> Rounding </rounding> [0..1]
      'Rounding direction and precision for amounts.'

   End Group: CommodityContent.model
</commodityForward>
```

Diagram



Schema Component Representation

```
<xsd:element name="commodityForward" type=" CommodityForward " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

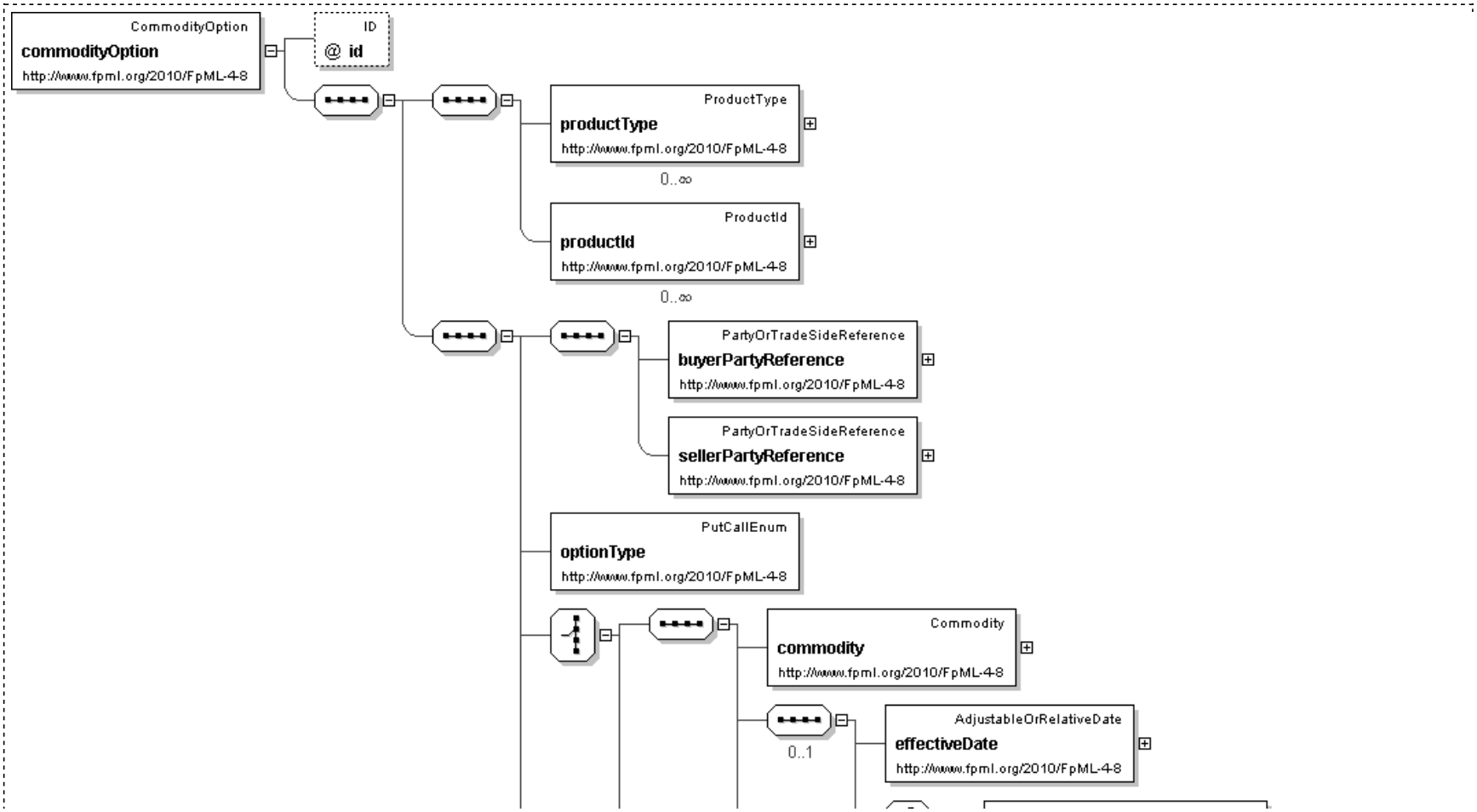
Element: commodityOption

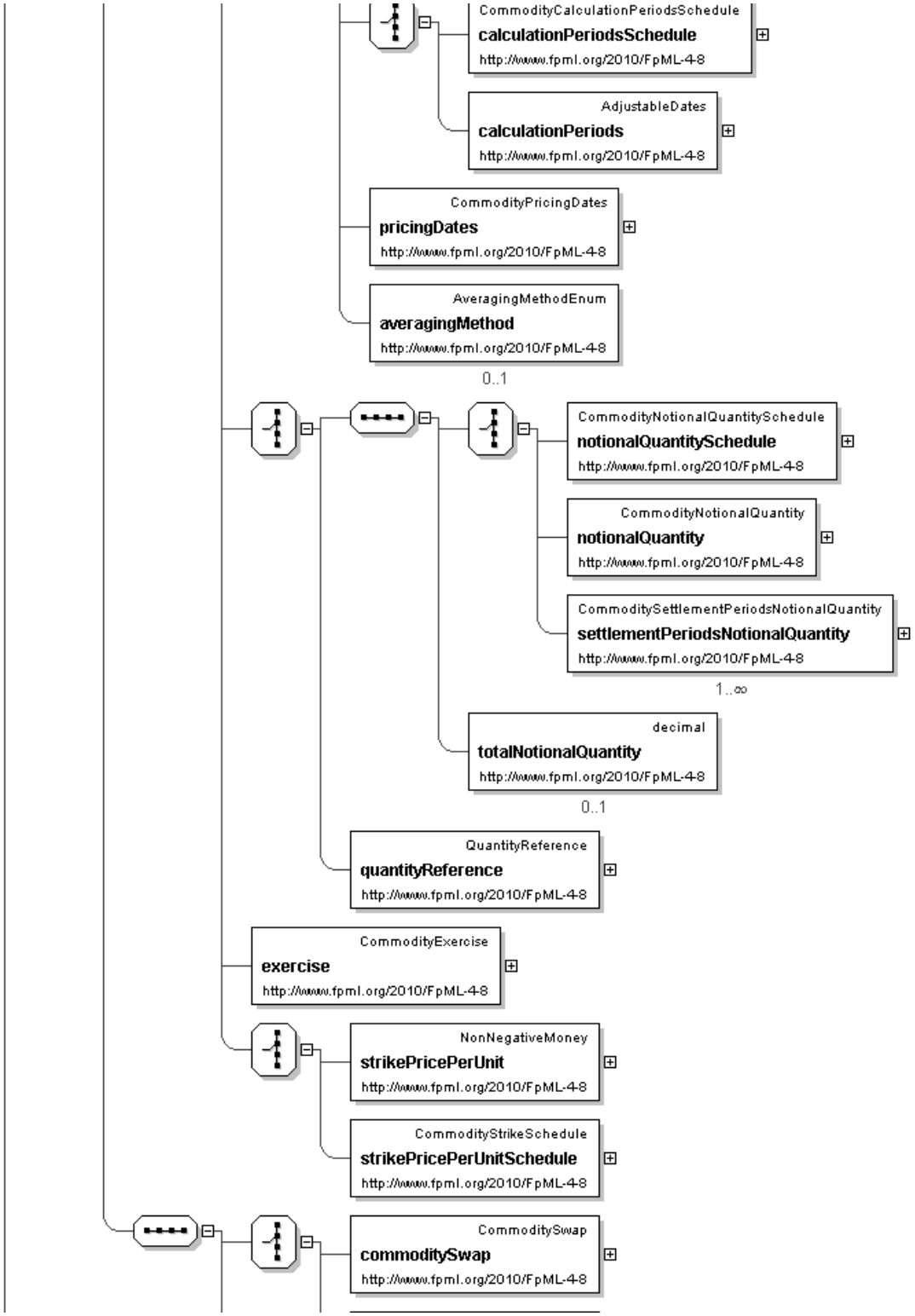
[Table of contents]

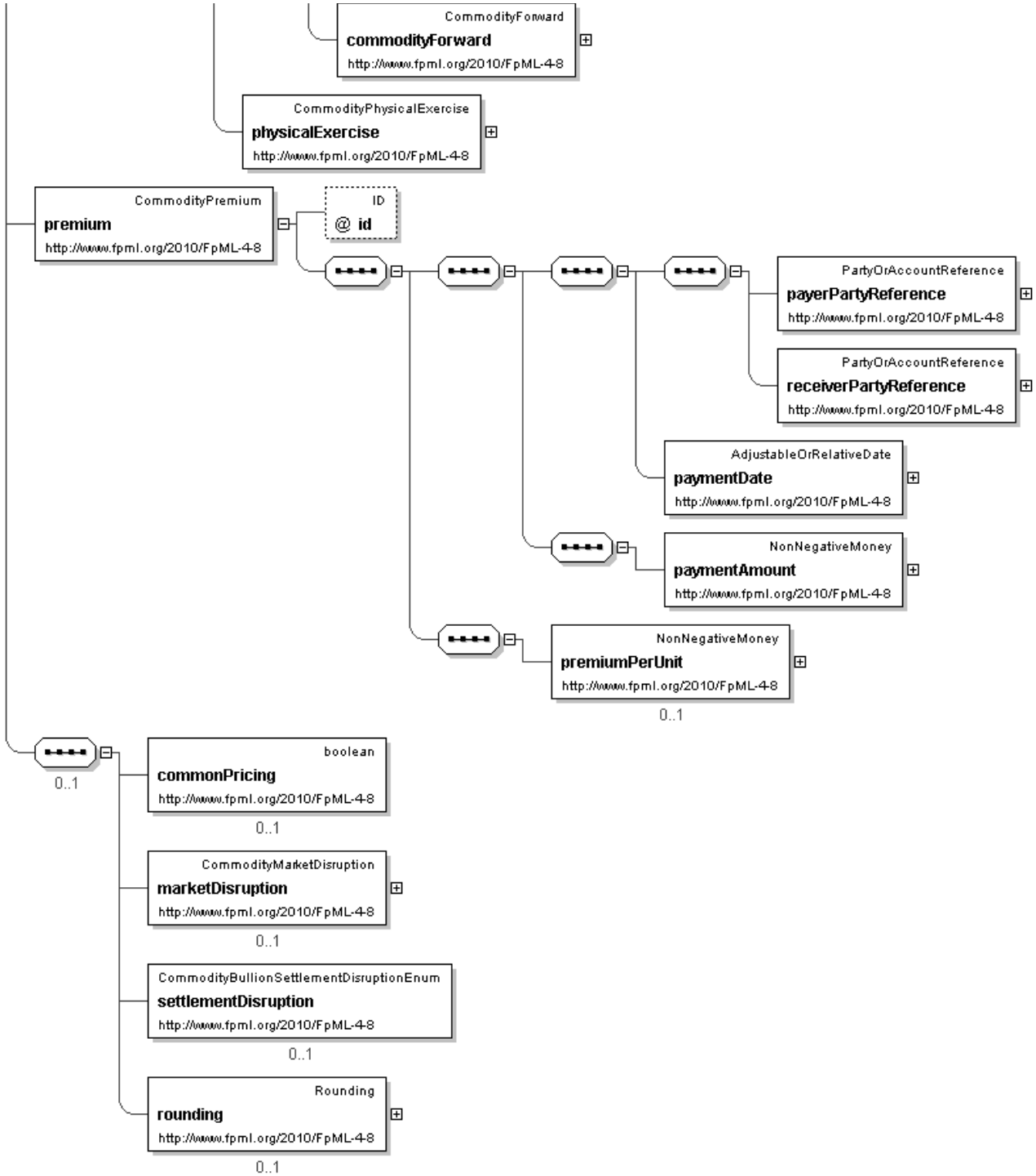
- *This element can be used wherever the following element is referenced:*
  - product

Name	commodityOption
Type	CommodityOption
Nillable	no
Abstract	no
Documentation	Defines a commodity option product.

Logical Diagram




CommodityCalculationPeriodsSchedule
calculationPeriodsSchedule
http://www.fpml.org/2010/FpML-4-8
AdjustableDates
calculationPeriods
http://www.fpml.org/2010/FpML-4-8
CommodityPricingDates
pricingDates
http://www.fpml.org/2010/FpML-4-8
AveragingMethodEnum
averagingMethod
http://www.fpml.org/2010/FpML-4-8
0..1
CommodityNotionalQuantitySchedule
notionalQuantitySchedule
http://www.fpml.org/2010/FpML-4-8
CommodityNotionalQuantity
notionalQuantity
http://www.fpml.org/2010/FpML-4-8
CommoditySettlementPeriodsNotionalQuantity
settlementPeriodsNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
1..∞
decimal
totalNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
0..1
QuantityReference
quantityReference
http://www.fpml.org/2010/FpML-4-8
CommodityExercise
exercise
http://www.fpml.org/2010/FpML-4-8
NonNegativeMoney
strikePricePerUnit
http://www.fpml.org/2010/FpML-4-8
CommodityStrikeSchedule
strikePricePerUnitSchedule
http://www.fpml.org/2010/FpML-4-8
CommoditySwap
commoditySwap
http://www.fpml.org/2010/FpML-4-8



XML Instance Representation

```
<commodityOption
```

id=" xsd:ID [0..1]">

<productType> ProductType </productType> [0..*]

'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

<productId> ProductId </productId> [0..*]

'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not strictly an enumerated list.'

<buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]

'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

<sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]

'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

<optionType> PutCallEnum </optionType> [1]

'The type of option transaction.'

Start Choice [1]

<commodity> Commodity </commodity> [1]

'Specifies the underlying component. At the time of the initial schema design, only underlyers of type Commodity are supported; the choice group in the future could offer the possibility of adding other types later.'

Start Group: CommodityAsian.model [0..1]

'A group containing properties specific to Asian options.'

<effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]

'The effective date of the Commodity Option Transaction. Note that the Termination/Expiration Date should be specified in expirationDate within the CommodityAmericanExercise type or the CommodityEuropeanExercise type, as applicable.'

Start Choice [1]

<calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule </calculationPeriodsSchedule> [1]

'A parametric representation of the Calculation Periods of the Commodity Option Transaction.'

<calculationPeriods> AdjustableDates </calculationPeriods> [1]

'An absolute representation of the Calculation Period start dates of the Commodity Option Transaction.'

End Choice

<pricingDates> CommodityPricingDates </pricingDates> [1]

'The dates on which the option will price.'

<averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]

'The Method of Averaging if there is more than one Pricing Date.'

End Group: CommodityAsian.model

Start Choice [1]

Start Choice [1]

<notionalQuantitySchedule> CommodityNotionalQuantitySchedule </notionalQuantitySchedule> [1]

'Allows the documentation of a shaped notional trade where the notional changes over the life of the transaction.'

<notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]

'The Notional Quantity.'

<settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity </settlementPeriodsNotionalQuantity> [1..*]

'For an electricity transaction, the Notional Quantity for a one or more groups of Settlement Periods to which the Notional Quantity is based. If the schedule differs for different groups of Settlement Periods, this element should be repeated.'

End Choice

<totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]

'The Total Notional Quantity.'

<quantityReference> QuantityReference </quantityReference> [1]

'A pointer style reference to a quantity defined on another leg.'

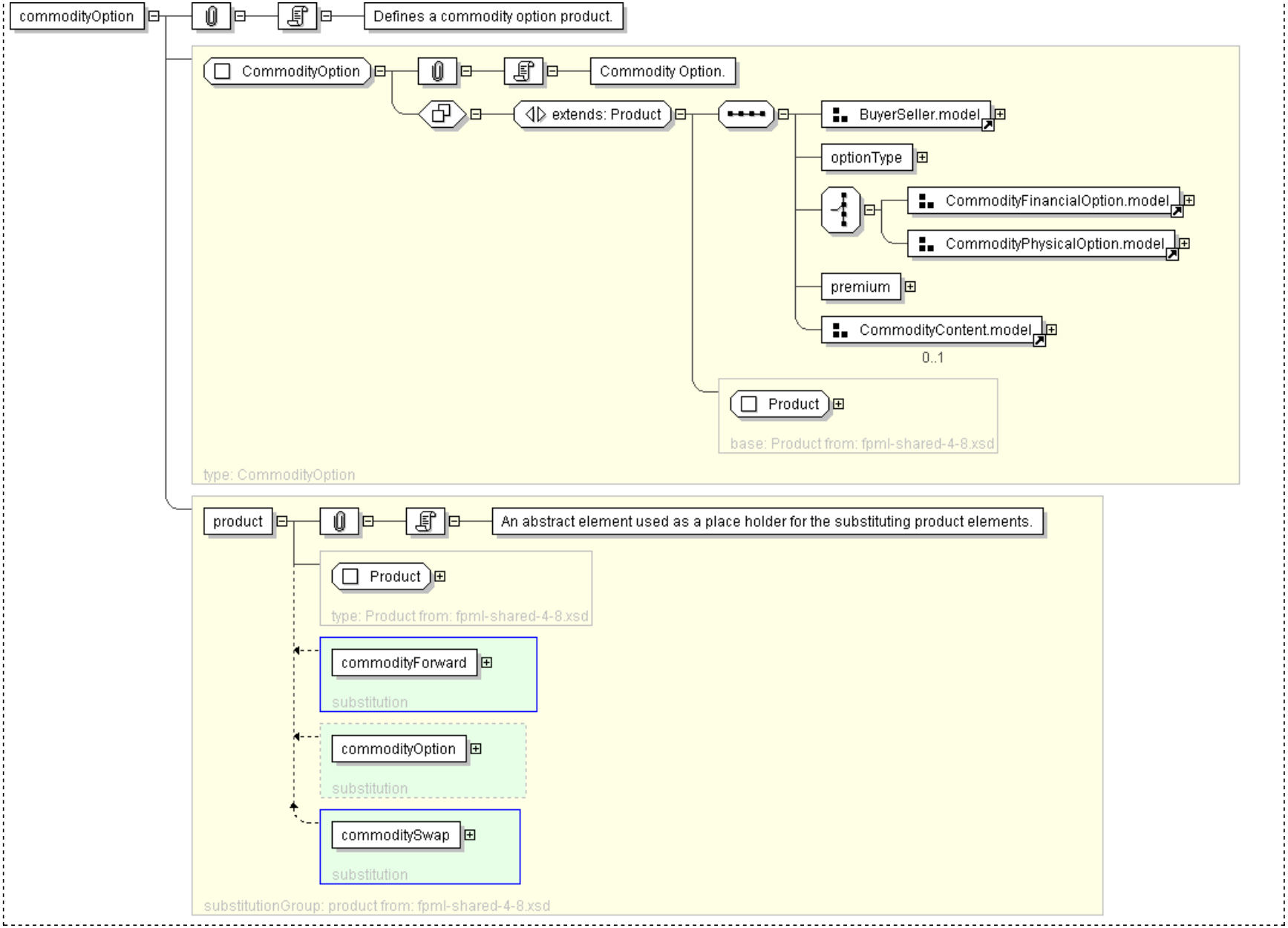
End Choice

<exercise> CommodityExercise </exercise> [1]

'The parameters for defining how the commodity option can be exercised and how it is settled.'

```
      Start Choice [1]
         <strikePricePerUnit> NonNegativeMoney </strikePricePerUnit> [1]
         'The currency amount of the strike price per unit.'

         <strikePricePerUnitSchedule> CommodityStrikeSchedule </strikePricePerUnitSchedule> [1]
      End Choice
      Start Choice [1]
         <commoditySwap> ... </commoditySwap> [1]
         <commodityForward> ... </commodityForward> [1]
      End Choice
      <physicalExercise> CommodityPhysicalExercise </physicalExercise> [1]
      'The parameters for defining how the commodity option can be exercised into a physical transaction.'

   End Choice
   <premium> CommodityPremium </premium> [1]
   'The option premium payable by the buyer to the seller.'

   Start Group: CommodityContent.model [0..1]
      <commonPricing> xsd:boolean </commonPricing> [0..1]
      'Common pricing may be relevant for a Transaction that references more than one Commodity Reference Price. If Common Pricing
      is not specified as applicable, it will be deemed not to apply.'

      <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
      'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA 2005 Commodity Definitions, as
      applicable.'

      <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption> [0..1]
      'The consequences of Bullion Settlement Disruption Events.'

      <rounding> Rounding </rounding> [0..1]
      'Rounding direction and precision for amounts.'

   End Group: CommodityContent.model
</commodityOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="commodityOption" type=" CommodityOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

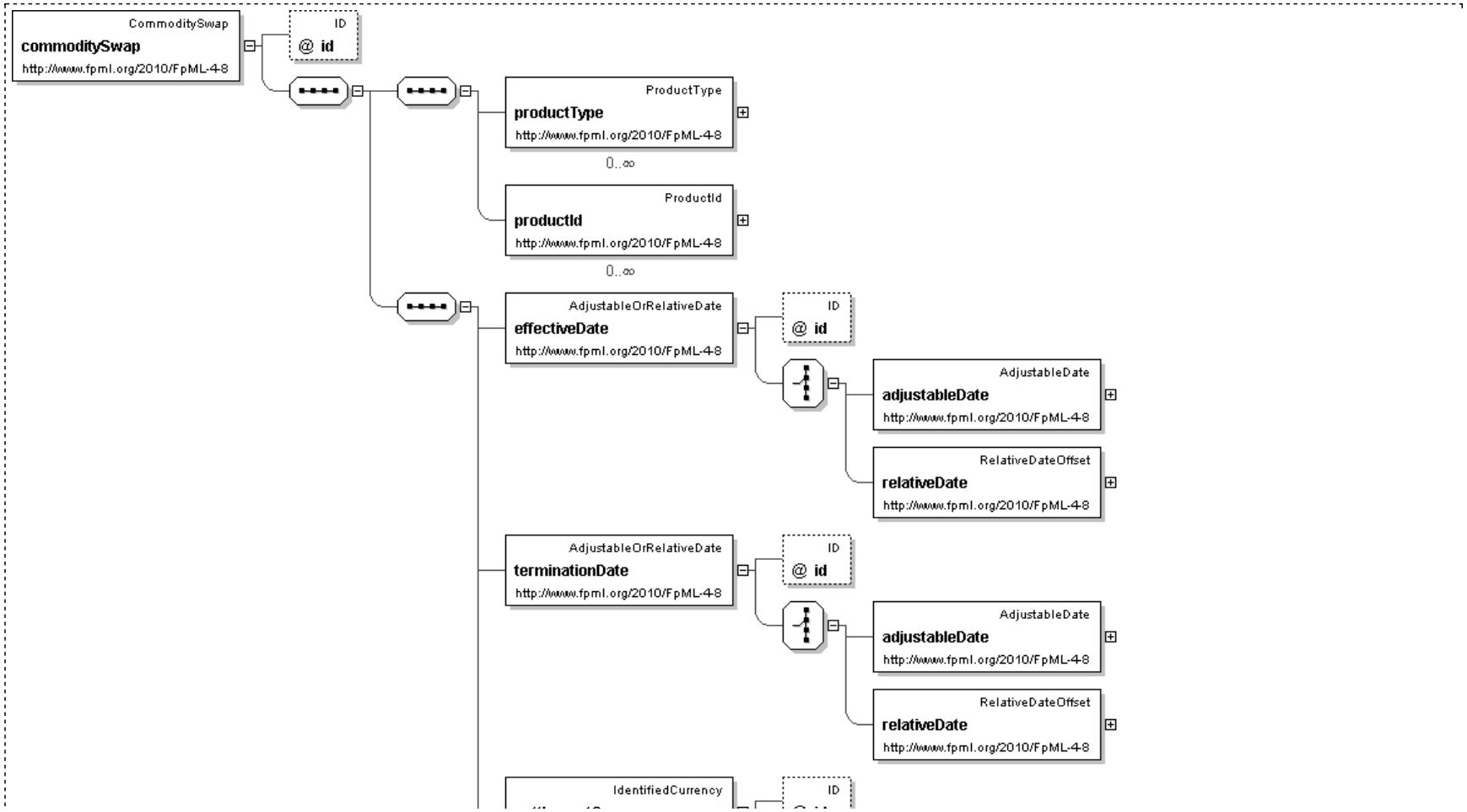
Element: commoditySwap

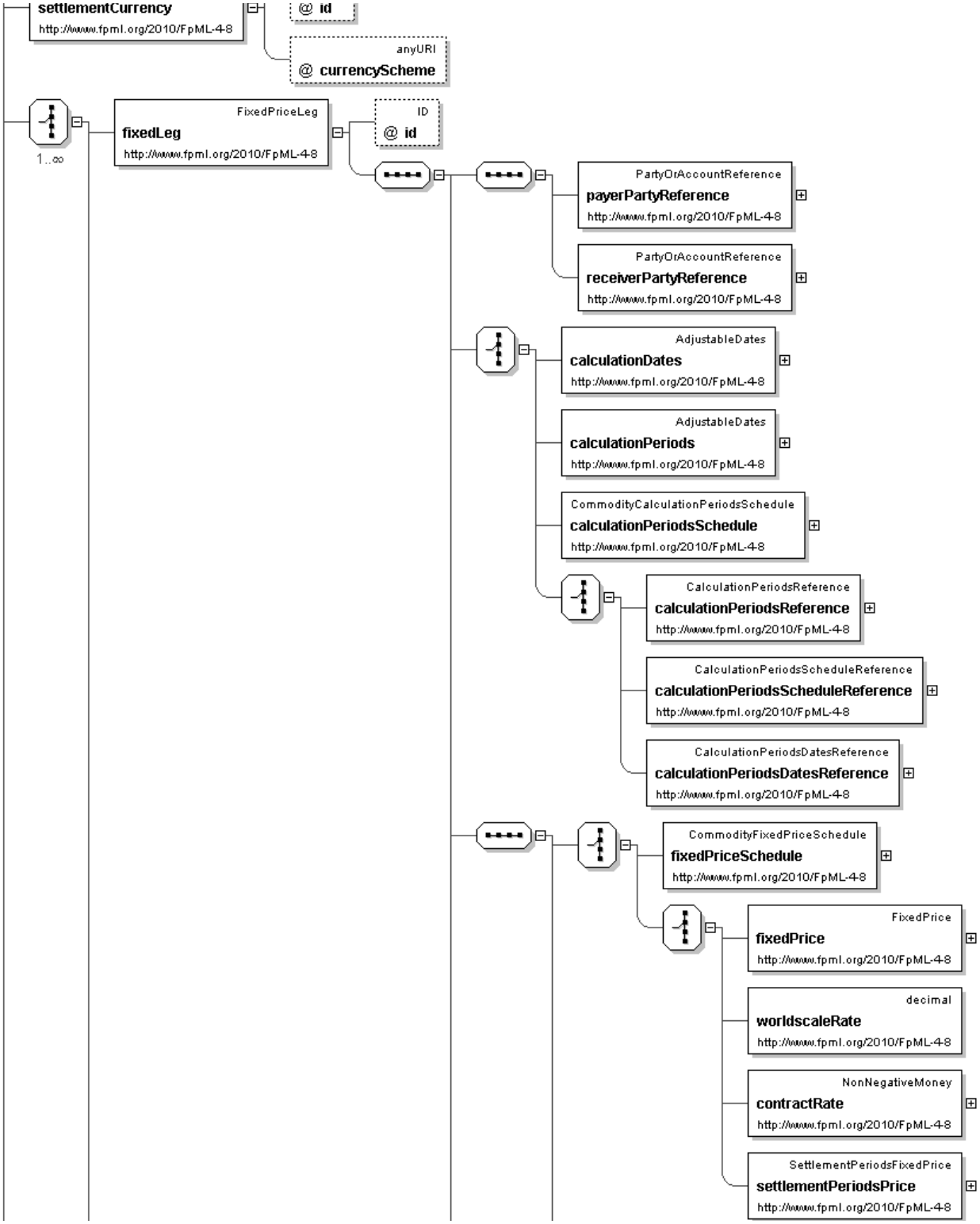
[Table of contents]

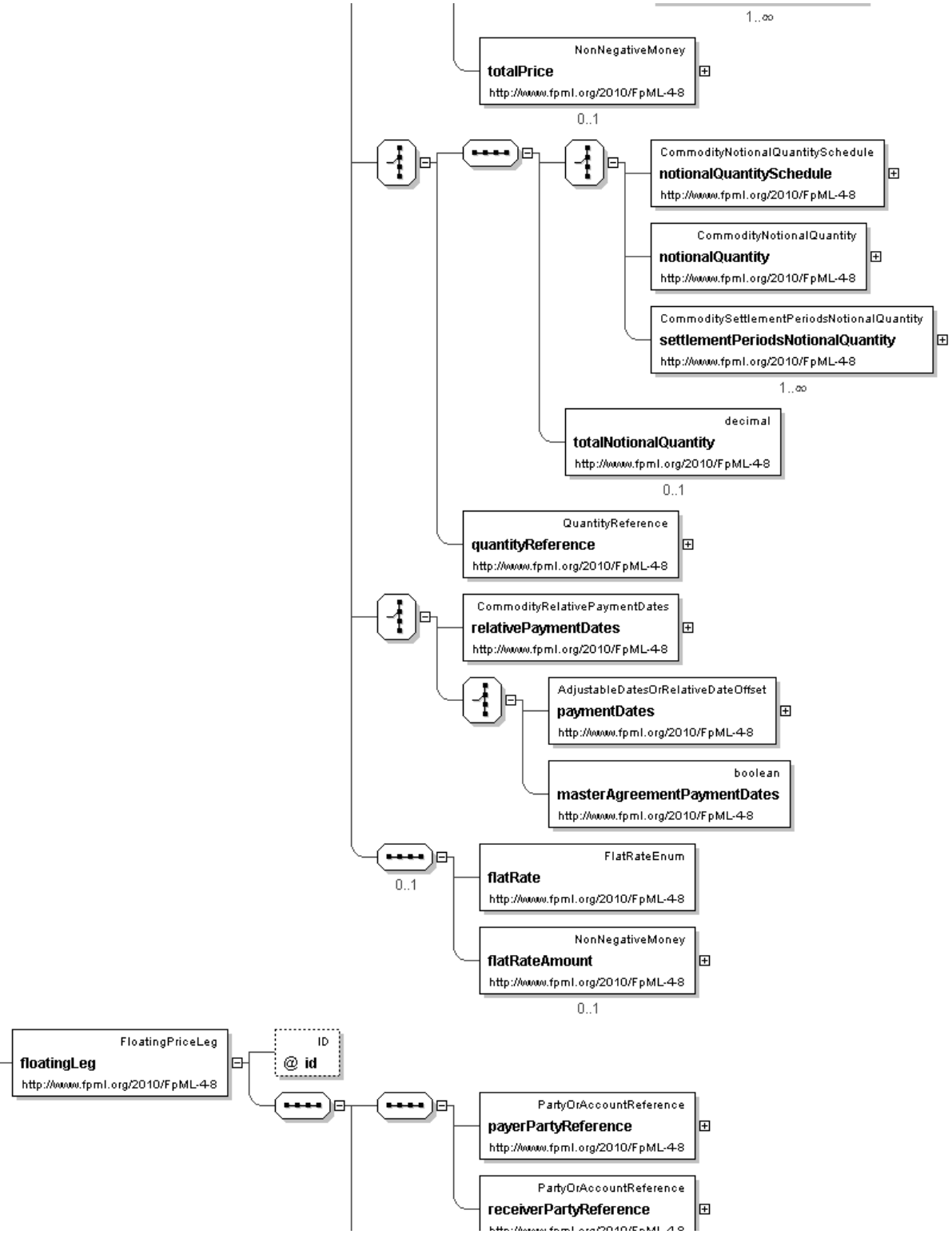
- *This element can be used wherever the following element is referenced:*
  - product

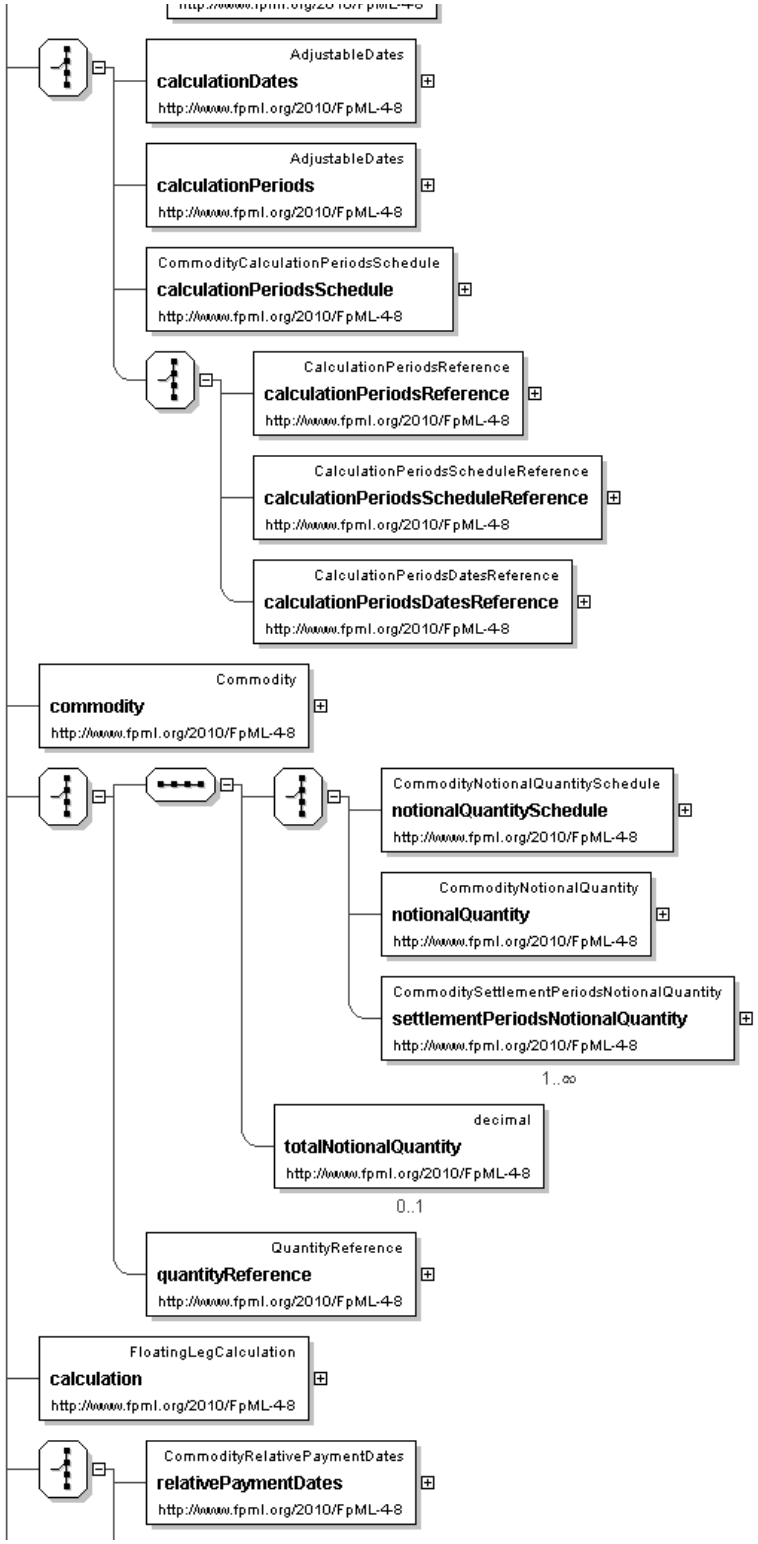
Name	commoditySwap
Used by (from the same schema document)	Model Group CommodityPhysicalOption.model
Type	CommoditySwap
Nillable	no
Abstract	no
Documentation	Defines a commodity swap product.

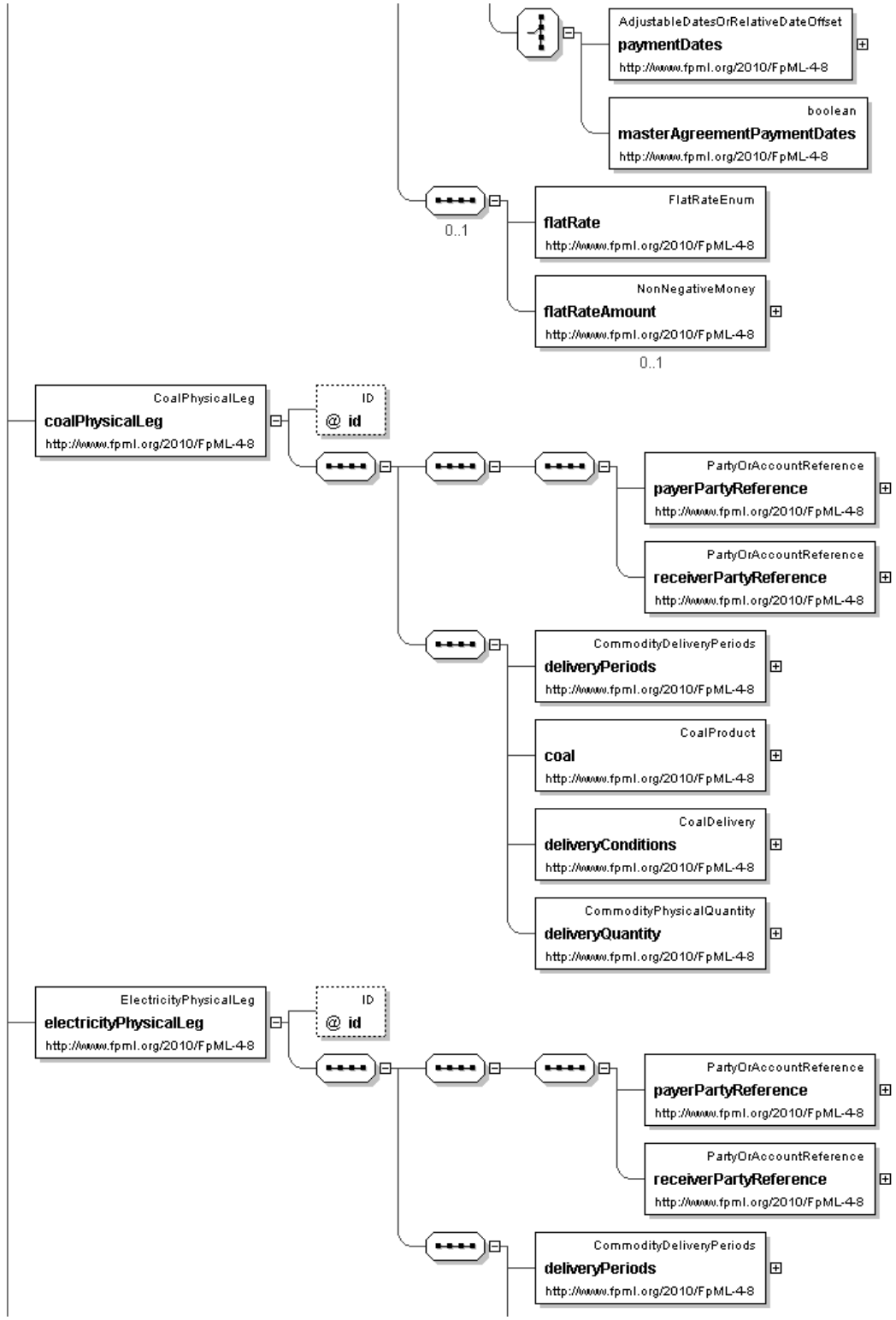
Logical Diagram

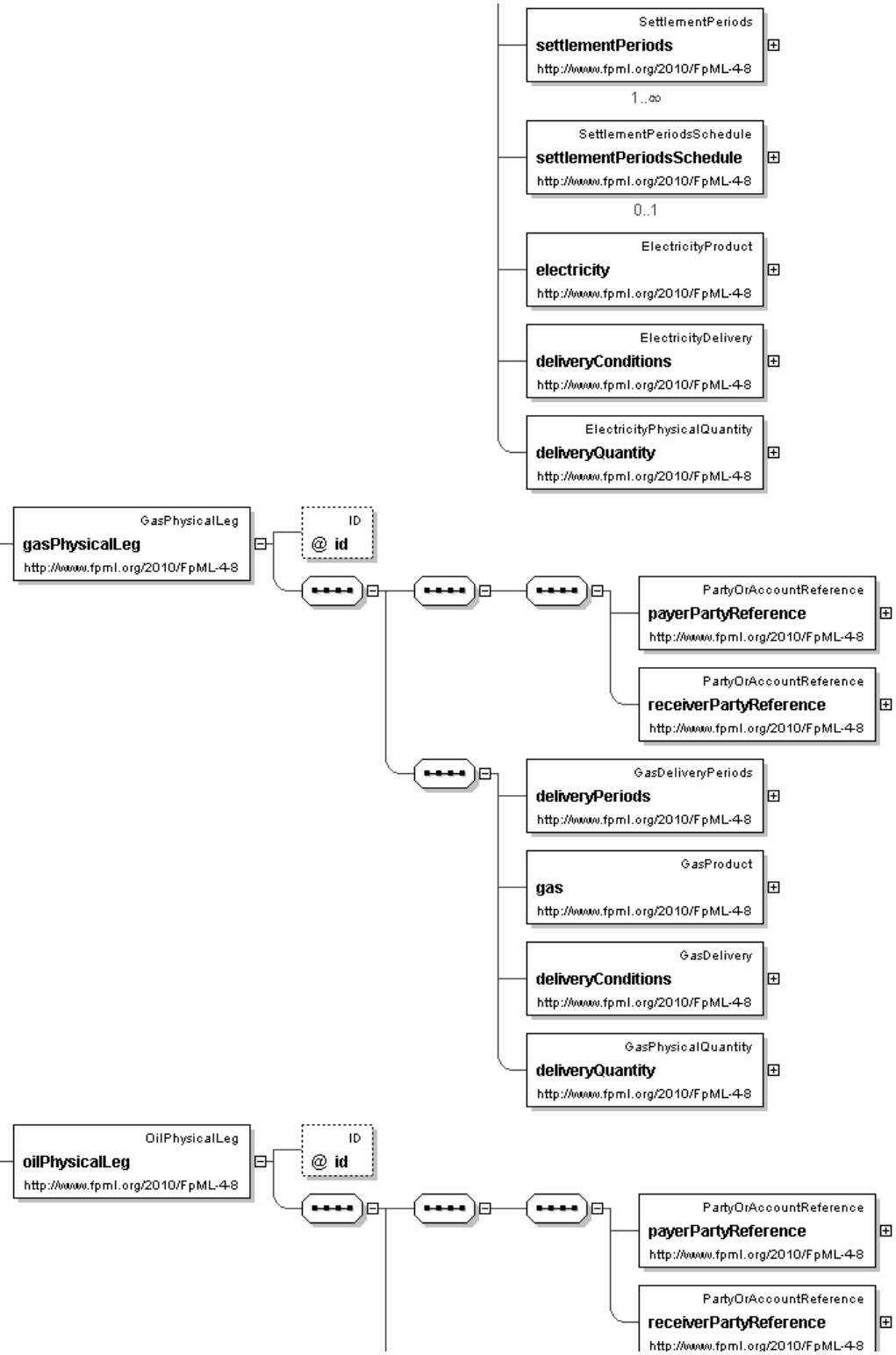



settlementCurrency
http://www.fpml.org/2010/FpML-4-8
@ id
anyURI
@ currencyScheme
1..∞
FixedPriceLeg
fixedLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
AdjustableDates
calculationDates
http://www.fpml.org/2010/FpML-4-8
AdjustableDates
calculationPeriods
http://www.fpml.org/2010/FpML-4-8
CommodityCalculationPeriodsSchedule
calculationPeriodsSchedule
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsReference
calculationPeriodsReference
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsScheduleReference
calculationPeriodsScheduleReference
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsDatesReference
calculationPeriodsDatesReference
http://www.fpml.org/2010/FpML-4-8
CommodityFixedPriceSchedule
fixedPriceSchedule
http://www.fpml.org/2010/FpML-4-8
FixedPrice
fixedPrice
http://www.fpml.org/2010/FpML-4-8
decimal
worldscaleRate
http://www.fpml.org/2010/FpML-4-8
NonNegativeMoney
contractRate
http://www.fpml.org/2010/FpML-4-8
SettlementPeriodsFixedPrice
settlementPeriodsPrice
http://www.fpml.org/2010/FpML-4-8


1..∞
NonNegativeMoney
totalPrice
http://www.fpml.org/2010/FpML-4-8
0..1
CommodityNotionalQuantitySchedule
notionalQuantitySchedule
http://www.fpml.org/2010/FpML-4-8
CommodityNotionalQuantity
notionalQuantity
http://www.fpml.org/2010/FpML-4-8
CommoditySettlementPeriodsNotionalQuantity
settlementPeriodsNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
1..∞
decimal
totalNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
0..1
QuantityReference
quantityReference
http://www.fpml.org/2010/FpML-4-8
CommodityRelativePaymentDates
relativePaymentDates
http://www.fpml.org/2010/FpML-4-8
AdjustableDatesOrRelativeDateOffset
paymentDates
http://www.fpml.org/2010/FpML-4-8
boolean
masterAgreementPaymentDates
http://www.fpml.org/2010/FpML-4-8
0..1
FlatRateEnum
flatRate
http://www.fpml.org/2010/FpML-4-8
NonNegativeMoney
flatRateAmount
http://www.fpml.org/2010/FpML-4-8
0..1
FloatingPriceLeg
floatingLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference


AdjustableDates
calculationDates
http://www.fpml.org/2010/FpML-4-8
AdjustableDates
calculationPeriods
http://www.fpml.org/2010/FpML-4-8
CommodityCalculationPeriodsSchedule
calculationPeriodsSchedule
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsReference
calculationPeriodsReference
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsScheduleReference
calculationPeriodsScheduleReference
http://www.fpml.org/2010/FpML-4-8
CalculationPeriodsDatesReference
calculationPeriodsDatesReference
http://www.fpml.org/2010/FpML-4-8
Commodity
commodity
http://www.fpml.org/2010/FpML-4-8
CommodityNotionalQuantitySchedule
notionalQuantitySchedule
http://www.fpml.org/2010/FpML-4-8
CommodityNotionalQuantity
notionalQuantity
http://www.fpml.org/2010/FpML-4-8
CommoditySettlementPeriodsNotionalQuantity
settlementPeriodsNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
1..∞
decimal
totalNotionalQuantity
http://www.fpml.org/2010/FpML-4-8
0..1
QuantityReference
quantityReference
http://www.fpml.org/2010/FpML-4-8
FloatingLegCalculation
calculation
http://www.fpml.org/2010/FpML-4-8
CommodityRelativePaymentDates
relativePaymentDates
http://www.fpml.org/2010/FpML-4-8


AdjustableDatesOrRelativeDateOffset
paymentDates
http://www.fpml.org/2010/FpML-4-8
boolean
masterAgreementPaymentDates
http://www.fpml.org/2010/FpML-4-8
0..1
FlatRateEnum
flatRate
http://www.fpml.org/2010/FpML-4-8
NonNegativeMoney
flatRateAmount
http://www.fpml.org/2010/FpML-4-8
0..1
CoalPhysicalLeg
coalPhysicalLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
CommodityDeliveryPeriods
deliveryPeriods
http://www.fpml.org/2010/FpML-4-8
CoalProduct
coal
http://www.fpml.org/2010/FpML-4-8
CoalDelivery
deliveryConditions
http://www.fpml.org/2010/FpML-4-8
CommodityPhysicalQuantity
deliveryQuantity
http://www.fpml.org/2010/FpML-4-8
ElectricityPhysicalLeg
electricityPhysicalLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
CommodityDeliveryPeriods
deliveryPeriods
http://www.fpml.org/2010/FpML-4-8

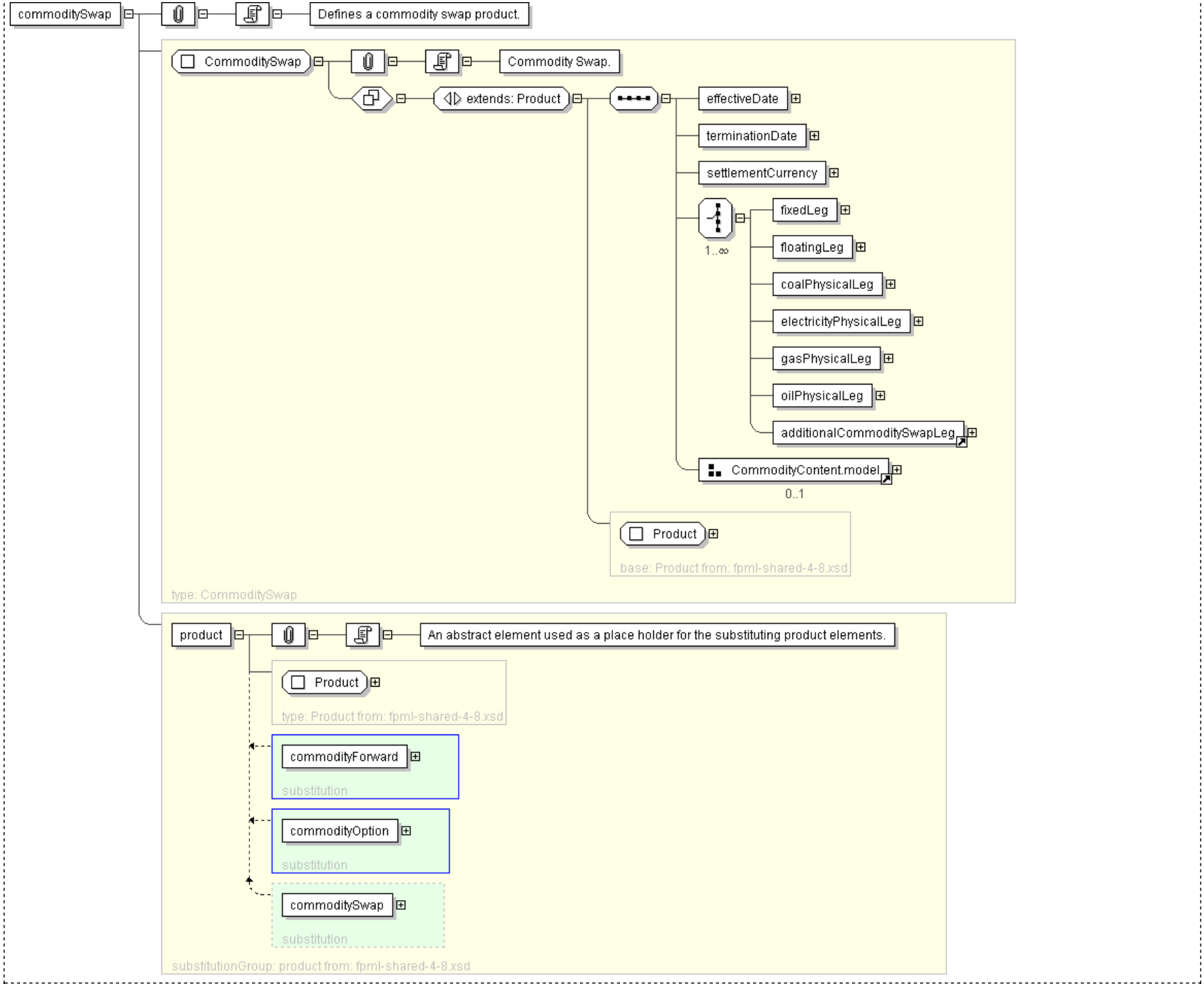

SettlementPeriods
settlementPeriods
http://www.fpml.org/2010/FpML-4-8
1..∞
SettlementPeriodsSchedule
settlementPeriodsSchedule
http://www.fpml.org/2010/FpML-4-8
0..1
ElectricityProduct
electricity
http://www.fpml.org/2010/FpML-4-8
ElectricityDelivery
deliveryConditions
http://www.fpml.org/2010/FpML-4-8
ElectricityPhysicalQuantity
deliveryQuantity
http://www.fpml.org/2010/FpML-4-8
GasPhysicalLeg
gasPhysicalLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
GasDeliveryPeriods
deliveryPeriods
http://www.fpml.org/2010/FpML-4-8
GasProduct
gas
http://www.fpml.org/2010/FpML-4-8
GasDelivery
deliveryConditions
http://www.fpml.org/2010/FpML-4-8
GasPhysicalQuantity
deliveryQuantity
http://www.fpml.org/2010/FpML-4-8
OilPhysicalLeg
oilPhysicalLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8

XML Instance Representation

```
<commoditySwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the
    domain values for this element are not strictly an enumerated list.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
    'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere else in the document
    (through the relativeDate component), this element will typically point to the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
    'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere else in the document
    (through the relativeDate component), this element will typically point to the termination date of the other leg of the swap.'

   <settlementCurrency> IdentifiedCurrency </settlementCurrency> [1]
    'The currency into which the Commodity Swap Transaction will settle. If this is not the same as the currency in which the Commodity
    Reference Price is quoted on a given floating leg of the Commodity Swap Transaction, then an FX rate should also be specified for that
    leg.'
```

```
Start Choice [1..*]
   <fixedLeg> FixedPriceLeg </fixedLeg> [1]
   'Fixed Price Leg.'

   <floatingLeg> FloatingPriceLeg </floatingLeg> [1]
   'Floating Price leg.'

   <coalPhysicalLeg> CoalPhysicalLeg </coalPhysicalLeg> [1]
   'Physically settled coal leg.'

   <electricityPhysicalLeg> ElectricityPhysicalLeg </electricityPhysicalLeg> [1]
   'Physically settled electricity leg.'

   <gasPhysicalLeg> GasPhysicalLeg </gasPhysicalLeg> [1]
   'Physically settled natural gas leg.'

   <oilPhysicalLeg> OilPhysicalLeg </oilPhysicalLeg> [1]
   'Physically settled oil or refined products leg.'

   <additionalCommoditySwapLeg> ... </additionalCommoditySwapLeg> [1]
End Choice
Start Group: CommodityContent.model [0..1]
   <commonPricing> xsd:boolean </commonPricing> [0..1]
   'Common pricing may be relevant for a Transaction that references more than one Commodity Reference Price. If Common Pricing is not
   specified as applicable, it will be deemed not to apply.'

   <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
   'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA 2005 Commodity Definitions, as applicable.'

   <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption> [0..1]
   'The consequences of Bullion Settlement Disruption Events.'

   <rounding> Rounding </rounding> [0..1]
   'Rounding direction and precision for amounts.'

End Group: CommodityContent.model
</commodity Swap>
```

Diagram



Schema Component Representation

```
<xsd:element name="commoditySwap" type=" CommoditySwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityAsian.model

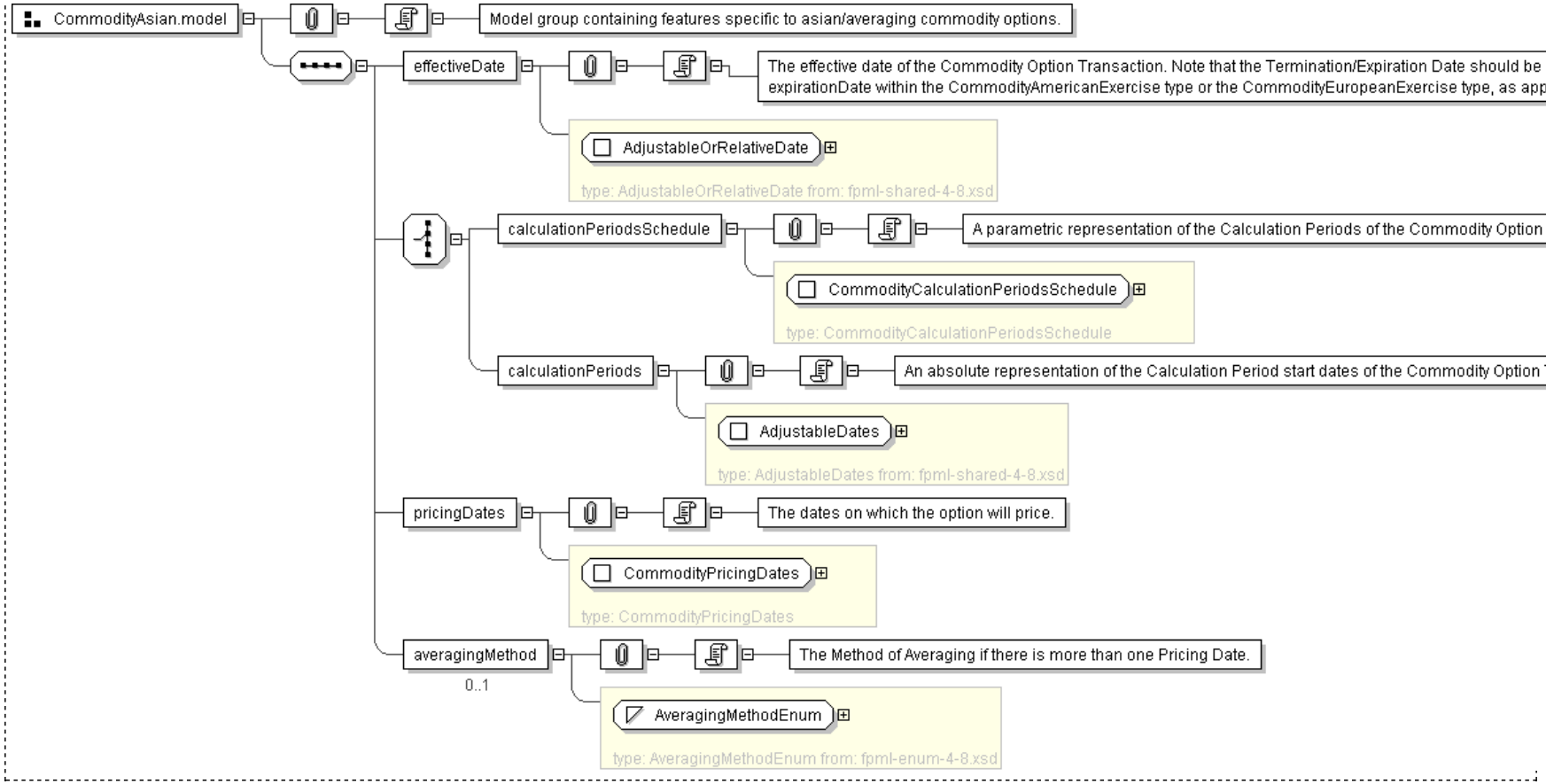
[Table of contents]

Name	CommodityAsian.model
Used by (from the same schema document)	Model Group CommodityFinancialOption.model
Documentation	Model group containing features specific to asian/averaging commodity options.

XML Instance Representation

```
<effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
'The effective date of the Commodity Option Transaction. Note that the Termination/Expiration Date should be specified in expirationDate
within the CommodityAmericanExercise type or the CommodityEuropeanExercise type, as applicable.'

Start Choice [1]
   <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule </calculationPeriodsSchedule> [1]
   'A parametric representation of the Calculation Periods of the Commodity Option Transaction.'

   <calculationPeriods> AdjustableDates </calculationPeriods> [1]
   'An absolute representation of the Calculation Period start dates of the Commodity Option Transaction.'

End Choice
<pricingDates> CommodityPricingDates </pricingDates> [1]
'The dates on which the option will price.'

<averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
'The Method of Averaging if there is more than one Pricing Date.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityAsian.model">
  <xsd:sequence>
    <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
    <xsd:choice>
      <xsd:element name="calculationPeriodsSchedule" type=" CommodityCalculationPeriodsSchedule "/>
      <xsd:element name="calculationPeriods" type=" AdjustableDates "/>
    </xsd:choice>
    <xsd:element name="pricingDates" type=" CommodityPricingDates "/>
    <xsd:element name="averagingMethod" type=" AveragingMethodEnum " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityCalculationPeriods.model

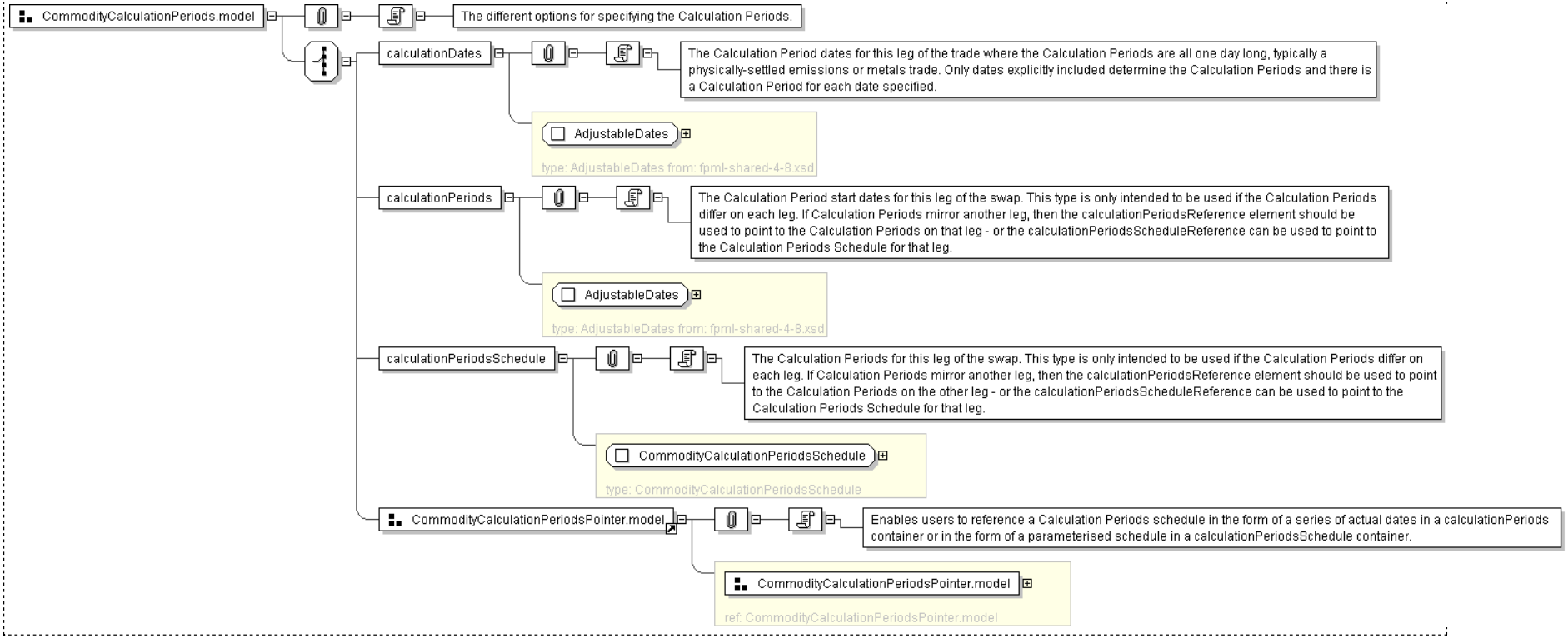
[Table of contents]

Name	CommodityCalculationPeriods.model
Used by (from the same schema document)	Complex Type FixedPriceLeg , Complex Type FloatingPriceLeg
Documentation	The different options for specifying the Calculation Periods.

XML Instance Representation

```
Start Choice [1]
   <calculationDates> AdjustableDates </calculationDates> [1]
   'The Calculation Period dates for this leg of the trade where the Calculation Periods are all one day long, typically a physically-settled emissions or metals trade.
   Only dates explicitly included determine the Calculation Periods and there is a Calculation Period for each date specified.'

   <calculationPeriods> AdjustableDates </calculationPeriods> [1]
   'The Calculation Period start dates for this leg of the swap. This type is only intended to be used if the Calculation Periods differ on each leg. If Calculation
   Periods mirror another leg, then the calculationPeriodsReference element should be used to point to the Calculation Periods on that leg - or the
   calculationPeriodsScheduleReference can be used to point to the Calculation Periods Schedule for that leg.'

   <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule </calculationPeriodsSchedule> [1]
   'The Calculation Periods for this leg of the swap. This type is only intended to be used if the Calculation Periods differ on each leg. If Calculation Periods mirror
   another leg, then the calculationPeriodsReference element should be used to point to the Calculation Periods on the other leg - or the
   calculationPeriodsScheduleReference can be used to point to the Calculation Periods Schedule for that leg.'

   Start Choice [1]
      <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

      <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

      <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

   End Choice
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityCalculationPeriods.model">
   <xsd:choice>
      <xsd:element name="calculationDates" type=" AdjustableDates "/>
      <xsd:element name="calculationPeriods" type=" AdjustableDates "/>
      <xsd:element name="calculationPeriodsSchedule" type=" CommodityCalculationPeriodsSchedule "/>
      <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityCalculationPeriodsPointer.model

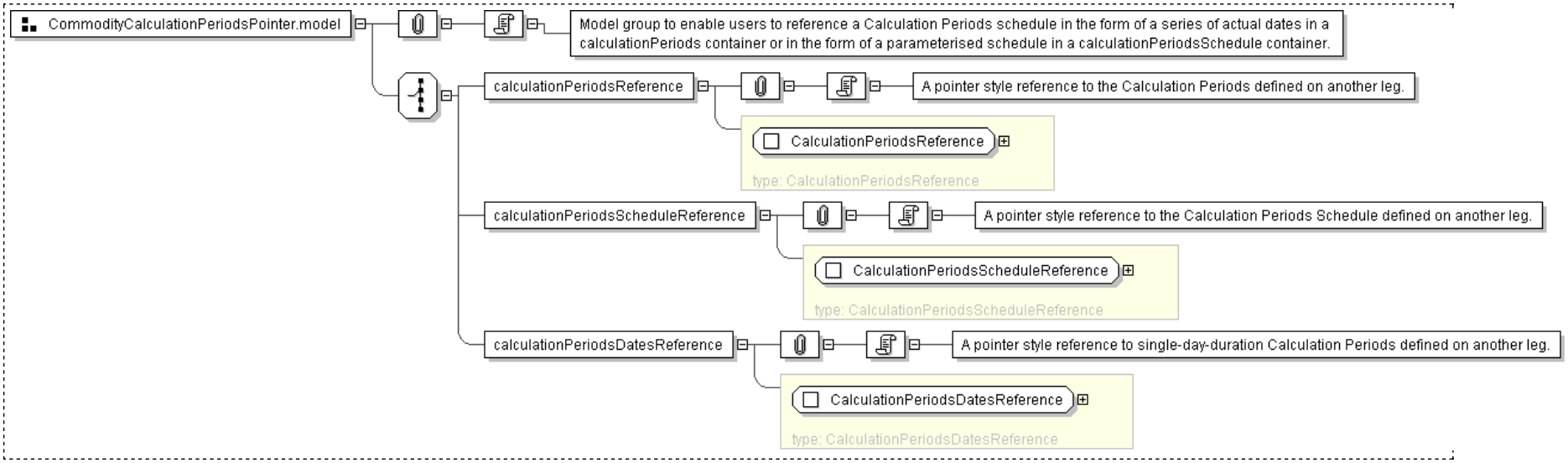
[Table of contents]

Name	CommodityCalculationPeriodsPointer.model
Used by (from the same schema document)	Complex Type CommodityDeliveryPeriods , Complex Type CommodityFixedPriceSchedule , Complex Type CommodityFx , Complex Type CommodityNotionalQuantitySchedule , Complex Type CommodityPricingDates , Complex Type CommodityRelativePaymentDates , Complex Type CommoditySpreadSchedule , Complex Type CommodityStrikeSchedule , Model Group CommodityCalculationPeriods.model
Documentation	Model group to enable users to reference a Calculation Periods schedule in the form of a series of actual dates in a calculationPeriods container or in the form of a parameterised schedule in a calculationPeriodsSchedule container.

XML Instance Representation

```
Start Choice [1]
  <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
  'A pointer style reference to the Calculation Periods defined on another leg.'

  <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
  'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

  <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
  'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityCalculationPeriodsPointer.model">
  <xsd:choice>
    <xsd:element name="calculationPeriodsReference" type=" CalculationPeriodsReference "/>
    <xsd:element name="calculationPeriodsScheduleReference" type=" CalculationPeriodsScheduleReference "/>
    <xsd:element name="calculationPeriodsDatesReference" type=" CalculationPeriodsDatesReference "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityCoalComposition.model

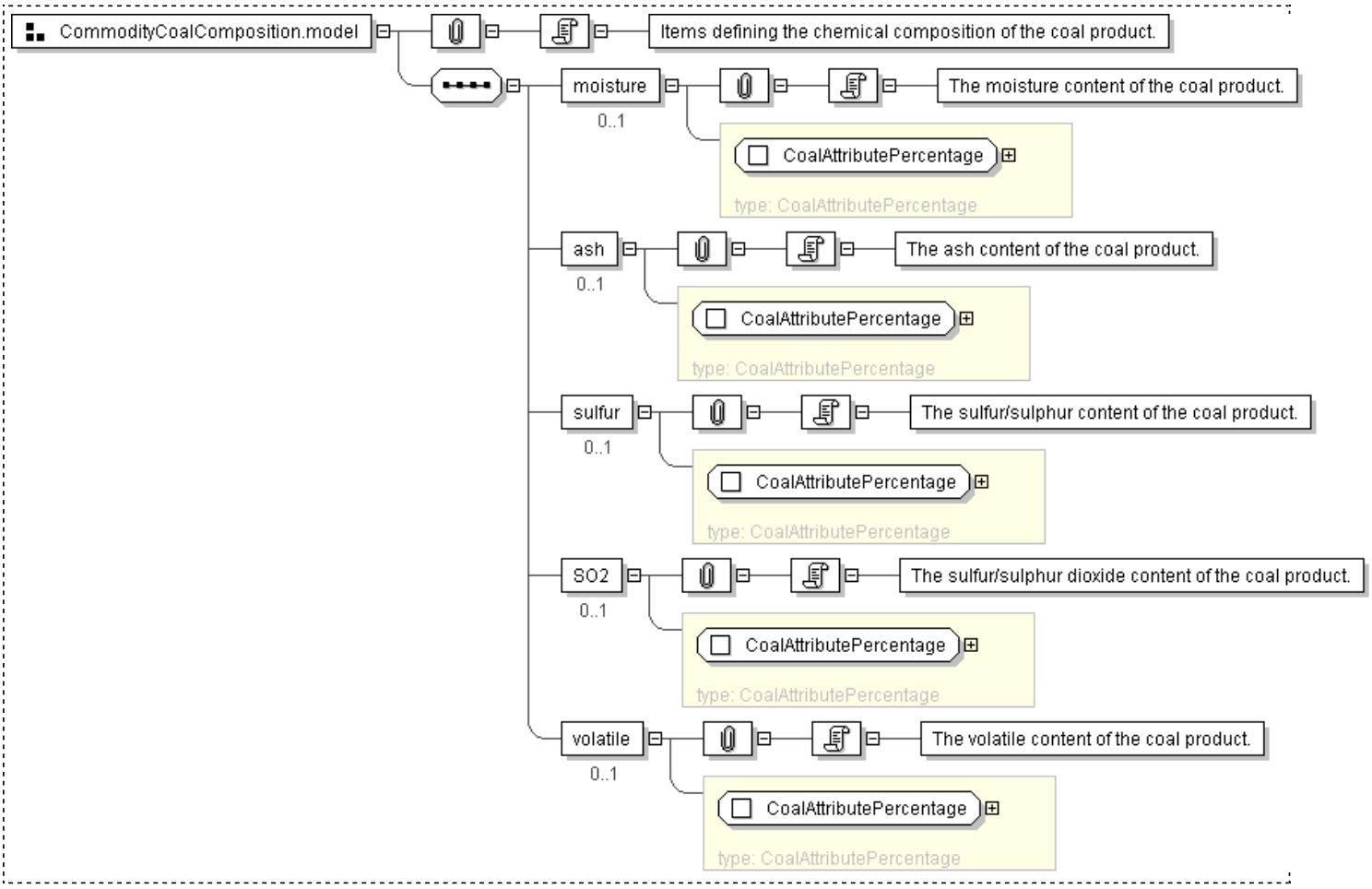
[Table of contents]

Name	CommodityCoalComposition.model
Used by (from the same schema document)	Complex Type CoalStandardQuality
Documentation	Items defining the chemical composition of the coal product.

XML Instance Representation

```
<moisture> CoalAttributePercentage </moisture> [0..1]
'The moisture content of the coal product.'

<ash> CoalAttributePercentage </ash> [0..1]
'The ash content of the coal product.'

<sulfur> CoalAttributePercentage </sulfur> [0..1]
'The sulfur/sulphur content of the coal product.'

<SO2> CoalAttributePercentage </SO2> [0..1]
'The sulfur/sulphur dioxide content of the coal product.'

<volatile> CoalAttributePercentage </volatile> [0..1]
'The volatile content of the coal product.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityCoalComposition.model">
  <xsd:sequence>
    <xsd:element name="moisture" type=" CoalAttributePercentage " minOccurs="0"/>
    <xsd:element name="ash" type=" CoalAttributePercentage " minOccurs="0"/>
```

```
    <xsd:element name="sulfur" type=" CoalAttributePercentage " minOccurs="0"/>
    <xsd:element name="SO2" type=" CoalAttributePercentage " minOccurs="0"/>
    <xsd:element name="volatile" type=" CoalAttributePercentage " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityCoalProperties.model

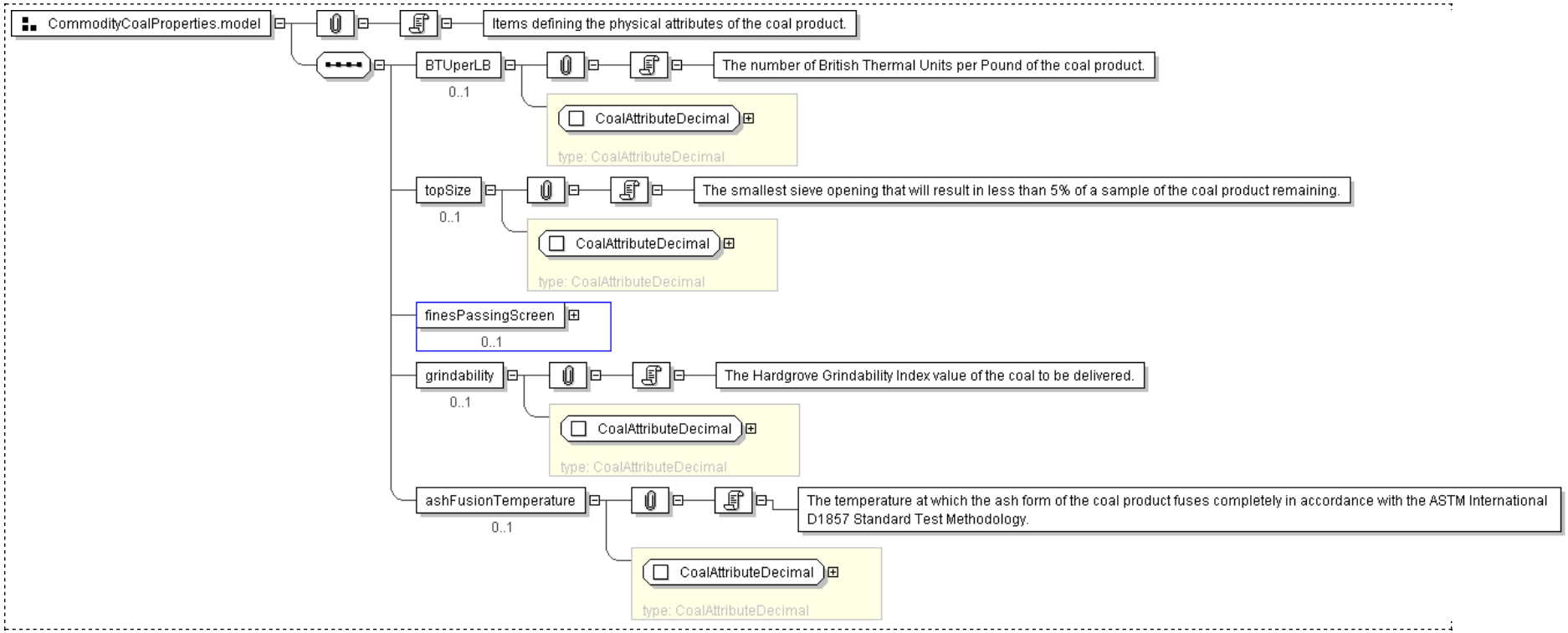
[Table of contents]

Name	CommodityCoalProperties.model
Used by (from the same schema document)	Complex Type CoalStandardQuality
Documentation	Items defining the physical attributes of the coal product.

XML Instance Representation

```
<BTUperLB> CoalAttributeDecimal </BTUperLB> [0..1]
'The number of British Thermal Units per Pound of the coal product.'

<topSize> CoalAttributeDecimal </topSize> [0..1]
'The smallest sieve opening that will result in less than 5% of a sample of the coal product remaining.'

<finesPassingScreen> CoalAttributeDecimal </finesPassingScreen> [0..1]
<grindability> CoalAttributeDecimal </grindability> [0..1]
'The Hardgrove Grindability Index value of the coal to be delivered.'

<ashFusionTemperature> CoalAttributeDecimal </ashFusionTemperature> [0..1]
'The temperature at which the ash form of the coal product fuses completely in accordance with the ASTM International D1857 Standard Test Methodology.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityCoalProperties.model">
  <xsd:sequence>
    <xsd:element name="BTUperLB" type=" CoalAttributeDecimal " minOccurs="0"/>
    <xsd:element name="topSize" type=" CoalAttributeDecimal " minOccurs="0"/>
    <xsd:element name="finesPassingScreen" type=" CoalAttributeDecimal " minOccurs="0"/>
```

```
      <xsd:element name="grindability" type=" CoalAttributeDecimal " minOccurs="0"/>
      <xsd:element name="ashFusionTemperature" type=" CoalAttributeDecimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityCoalReducingAtmosphere.model

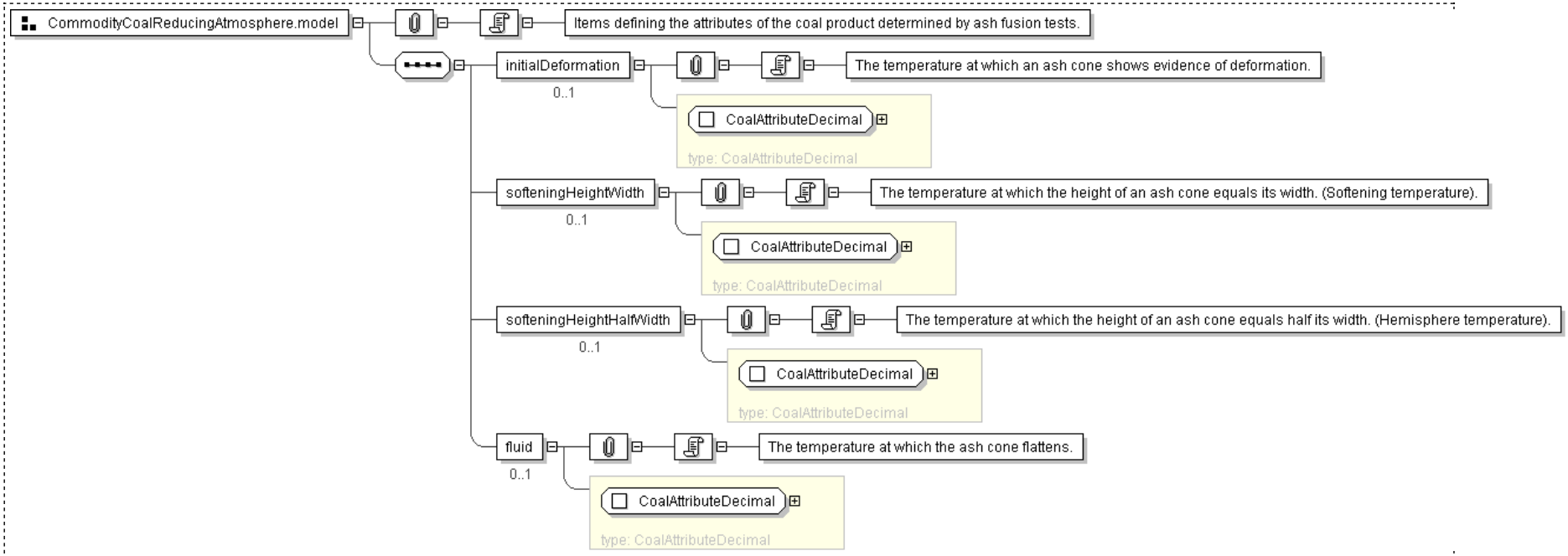
[Table of contents]

Name	CommodityCoalReducingAtmosphere.model
Used by (from the same schema document)	Complex Type CoalStandardQuality
Documentation	Items defining the attributes of the coal product determined by ash fusion tests.

XML Instance Representation

```
<initialDeformation> CoalAttributeDecimal </initialDeformation> [0..1]
'The temperature at which an ash cone shows evidence of deformation.'

<softeningHeightWidth> CoalAttributeDecimal </softeningHeightWidth> [0..1]
'The temperature at which the height of an ash cone equals its width. (Softening temperature).'

<softeningHeightHalfWidth> CoalAttributeDecimal </softeningHeightHalfWidth> [0..1]
'The temperature at which the height of an ash cone equals half its width. (Hemisphere temperature).'

<fluid> CoalAttributeDecimal </fluid> [0..1]
'The temperature at which the ash cone flattens.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityCoalReducingAtmosphere.model">
  <xsd:sequence>
    <xsd:element name="initialDeformation" type=" CoalAttributeDecimal " minOccurs="0"/>
    <xsd:element name="softeningHeightWidth" type=" CoalAttributeDecimal " minOccurs="0"/>
    <xsd:element name="softeningHeightHalfWidth" type=" CoalAttributeDecimal " minOccurs="0"/>
    <xsd:element name="fluid" type=" CoalAttributeDecimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityContent.model

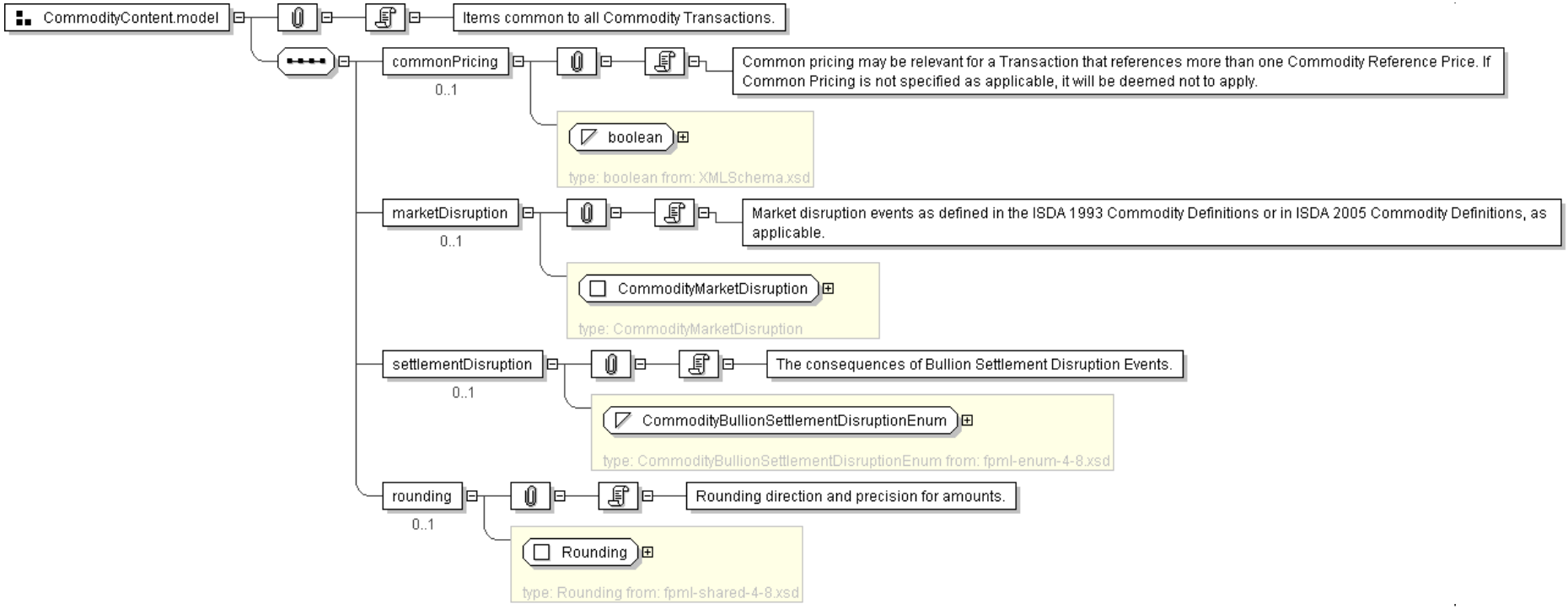
[Table of contents]

Name	CommodityContent.model
Used by (from the same schema document)	Complex Type CommodityForward , Complex Type CommodityOption , Complex Type CommoditySwap
Documentation	Items common to all Commodity Transactions.

XML Instance Representation

```
<commonPricing> xsd:boolean </commonPricing> [0..1]
'Common pricing may be relevant for a Transaction that references more than one Commodity Reference Price. If Common Pricing is not specified as applicable, it will be deemed not to apply.'

<marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA 2005 Commodity Definitions, as applicable.'

<settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption> [0..1]
'The consequences of Bullion Settlement Disruption Events.'

<rounding> Rounding </rounding> [0..1]
'Rounding direction and precision for amounts.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityContent.model">
  <xsd:sequence>
    <xsd:element name="commonPricing" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="marketDisruption" type=" CommodityMarketDisruption " minOccurs="0"/>
```

```
      <xsd:element name="settlementDisruption" type=" CommodityBullionSettlementDisruptionEnum " minOccurs="0"/>
      <xsd:element name="rounding" type=" Rounding " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityDeliveryPeriodsPointer.model

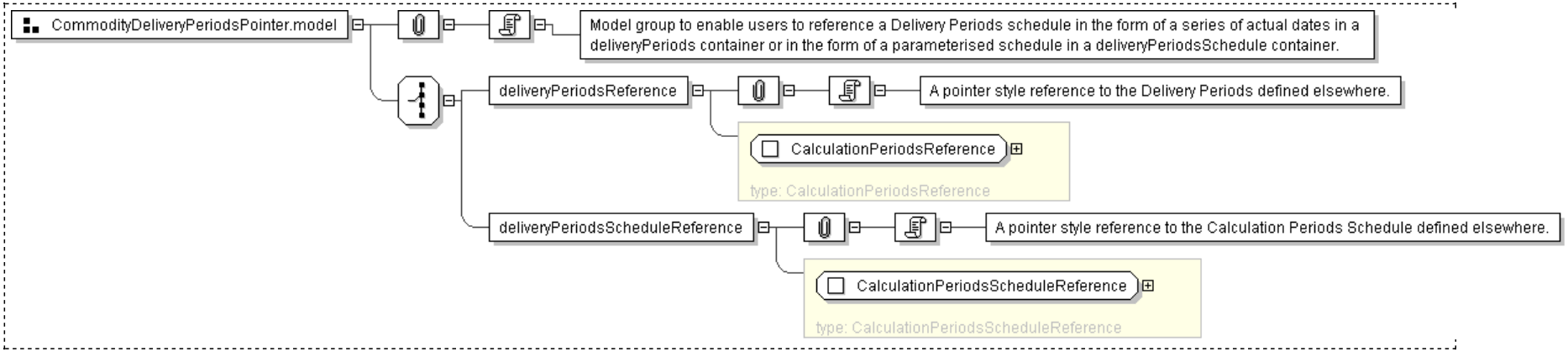
[Table of contents]

Name	CommodityDeliveryPeriodsPointer.model
Used by (from the same schema document)	Complex Type CoalStandardQualitySchedule , Complex Type CommodityPhysicalQuantitySchedule , Complex Type SettlementPeriodsSchedule
Documentation	Model group to enable users to reference a Delivery Periods schedule in the form of a series of actual dates in a deliveryPeriods container or in the form of a parameterised schedule in a deliveryPeriodsSchedule container.

XML Instance Representation

```
Start Choice [1]
   <deliveryPeriodsReference> CalculationPeriodsReference </deliveryPeriodsReference> [1]
   'A pointer style reference to the Delivery Periods defined elsewhere.'

   <deliveryPeriodsScheduleReference> CalculationPeriodsScheduleReference </deliveryPeriodsScheduleReference> [1]
   'A pointer style reference to the Calculation Periods Schedule defined elsewhere.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityDeliveryPeriodsPointer.model">
  <xsd:choice>
    <xsd:element name="deliveryPeriodsReference" type=" CalculationPeriodsReference "/>
    <xsd:element name="deliveryPeriodsScheduleReference" type=" CalculationPeriodsScheduleReference "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityDeliveryPoints.model

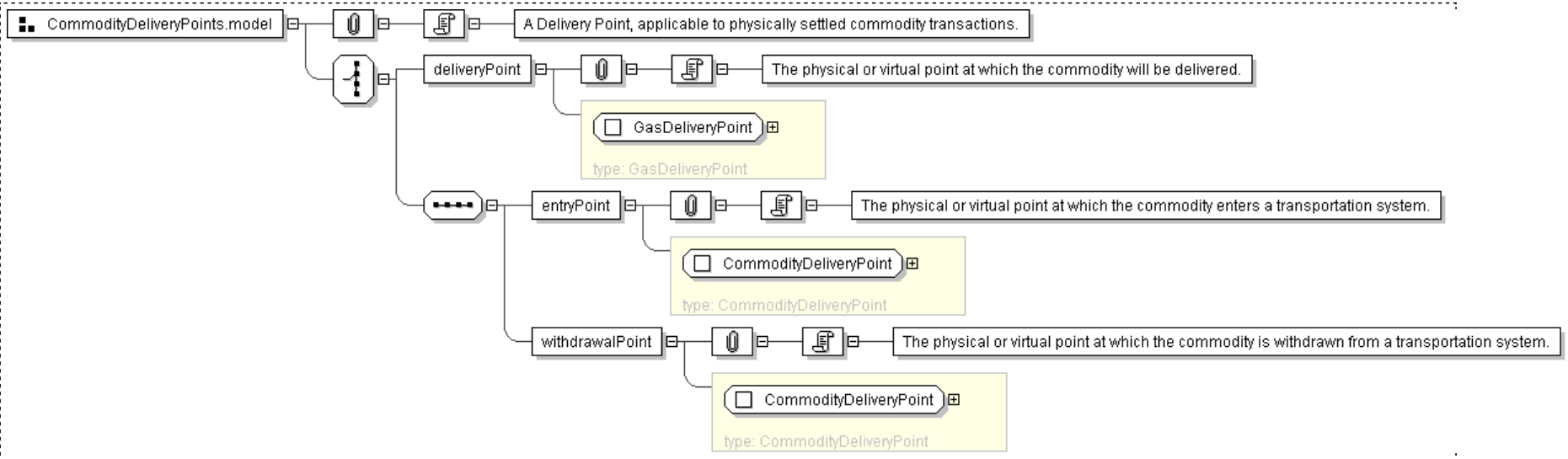
[Table of contents]

Name	CommodityDeliveryPoints.model
Used by (from the same schema document)	Complex Type GasDelivery
Documentation	A Delivery Point, applicable to physically settled commodity transactions.

XML Instance Representation

```
Start Choice [1]
   <deliveryPoint> GasDeliveryPoint </deliveryPoint> [1]
   'The physical or virtual point at which the commodity will be delivered.'

   <entryPoint> CommodityDeliveryPoint </entryPoint> [1]
   'The physical or virtual point at which the commodity enters a transportation system.'

   <withdrawalPoint> CommodityDeliveryPoint </withdrawalPoint> [1]
   'The physical or virtual point at which the commodity is withdrawn from a transportation system.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityDeliveryPoints.model">
   <xsd:choice>
      <xsd:element name="deliveryPoint" type=" GasDeliveryPoint "/>
      <xsd:sequence>
         <xsd:element name="entryPoint" type=" CommodityDeliveryPoint "/>
         <xsd:element name="withdrawalPoint" type=" CommodityDeliveryPoint "/>
      </xsd:sequence>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityFinancialOption.model

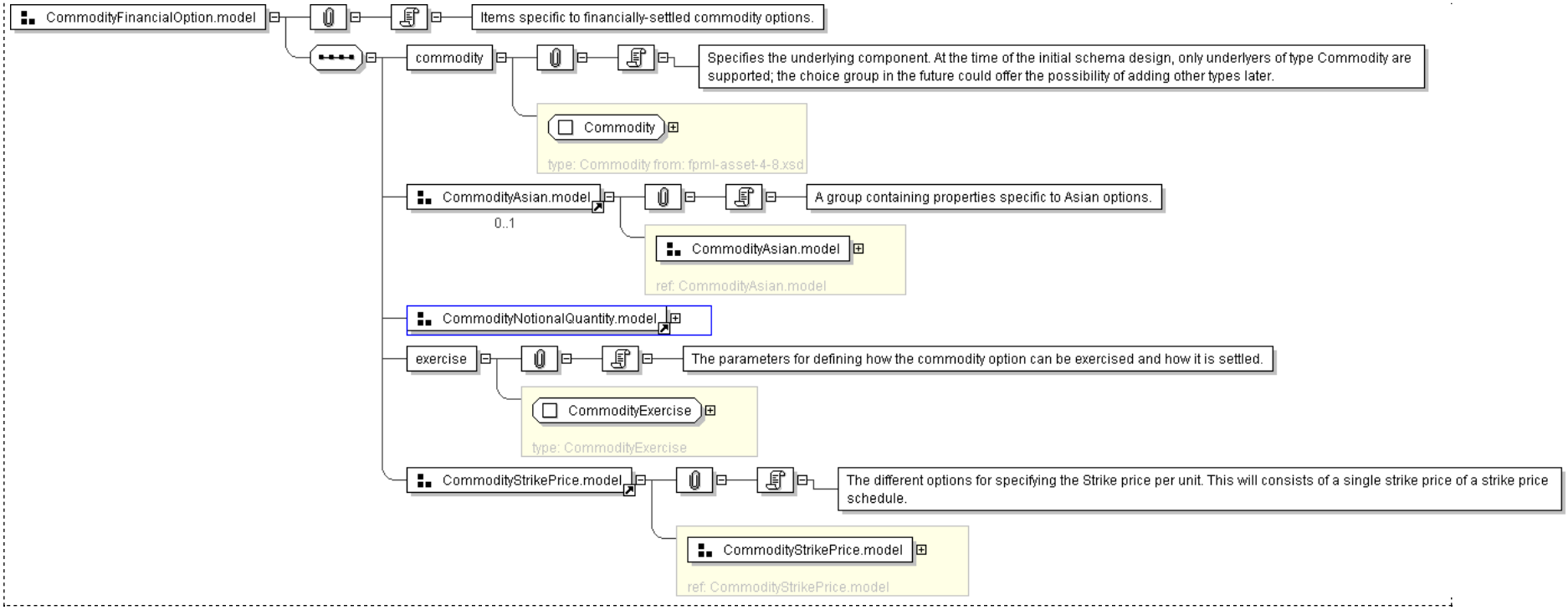
[Table of contents]

Name	CommodityFinancialOption.model
Used by (from the same schema document)	Complex Type CommodityOption
Documentation	Items specific to financially-settled commodity options.

XML Instance Representation

```
<commodity> Commodity </commodity> [1]
 'Specifies the underlying component. At the time of the initial schema design, only underlyers of type Commodity are supported; the choice group in the
 future could offer the possibility of adding other types later.'

Start Group: CommodityAsian.model [0..1]
 'A group containing properties specific to Asian options.'


   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'The effective date of the Commodity Option Transaction. Note that the Termination/Expiration Date should be specified in expirationDate within the
   CommodityAmericanExercise type or the CommodityEuropeanExercise type, as applicable.'

   Start Choice [1]
      <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule </calculationPeriodsSchedule> [1]
      'A parametric representation of the Calculation Periods of the Commodity Option Transaction.'

      <calculationPeriods> AdjustableDates </calculationPeriods> [1]
      'An absolute representation of the Calculation Period start dates of the Commodity Option Transaction.'

   End Choice
   <pricingDates> CommodityPricingDates </pricingDates> [1]
   'The dates on which the option will price.'

   <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
   'The Method of Averaging if there is more than one Pricing Date.'

End Group: CommodityAsian.model
Start Choice [1]
   Start Choice [1]
      <notionalQuantitySchedule> CommodityNotionalQuantitySchedule </notionalQuantitySchedule> [1]
      'Allows the documentation of a shaped notional trade where the notional changes over the life of the transaction.'

      <notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]
      'The Notional Quantity.'

      <settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity </settlementPeriodsNotionalQuantity> [1..*]
      'For an electricity transaction, the Notional Quantity for a one or more groups of Settlement Periods to which the Notional Quantity is based. If the
      schedule differs for different groups of Settlement Periods, this element should be repeated.'

   End Choice
   <totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]
   'The Total Notional Quantity.'

   <quantityReference> QuantityReference </quantityReference> [1]
   'A pointer style reference to a quantity defined on another leg.'

End Choice
<exercise> CommodityExercise </exercise> [1]
 'The parameters for defining how the commodity option can be exercised and how it is settled.'

Start Choice [1]
   <strikePricePerUnit> NonNegativeMoney </strikePricePerUnit> [1]
   'The currency amount of the strike price per unit.'

   <strikePricePerUnitSchedule> CommodityStrikeSchedule </strikePricePerUnitSchedule> [1]
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityFinancialOption.model">
  <xsd:sequence>
    <xsd:element name="commodity" type=" Commodity "/>
    <xsd:group ref=" CommodityAsian.model " minOccurs="0"/>
    <xsd:group ref=" CommodityNotionalQuantity.model "/>
    <xsd:element name="exercise" type=" CommodityExercise "/>
    <xsd:group ref=" CommodityStrikePrice.model "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityFixedPhysicalQuantity.model

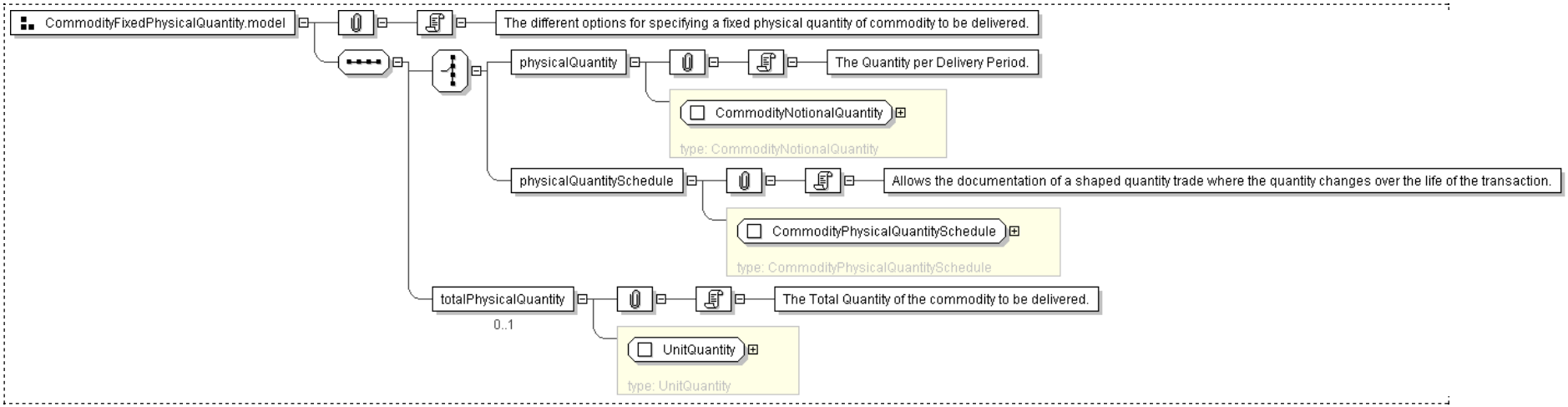
[Table of contents]

Name	CommodityFixedPhysicalQuantity.model
Used by (from the same schema document)	Complex Type BullionPhysicalLeg , Complex Type CommodityPhysicalQuantity , Complex Type GasPhysicalQuantity
Documentation	The different options for specifying a fixed physical quantity of commodity to be delivered.

XML Instance Representation

```
Start Choice [1]
   <physicalQuantity> CommodityNotionalQuantity </physicalQuantity> [1]
   'The Quantity per Delivery Period.'

   <physicalQuantitySchedule> CommodityPhysicalQuantitySchedule </physicalQuantitySchedule> [1]
   'Allows the documentation of a shaped quantity trade where the quantity changes over the life of the transaction.'

End Choice
<totalPhysicalQuantity> UnitQuantity </totalPhysicalQuantity> [0..1]
'The Total Quantity of the commodity to be delivered.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityFixedPhysicalQuantity.model">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="physicalQuantity" type=" CommodityNotionalQuantity "/>
         <xsd:element name="physicalQuantitySchedule" type=" CommodityPhysicalQuantitySchedule "/>
      </xsd:choice>
      <xsd:element name="totalPhysicalQuantity" type=" UnitQuantity " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityFixedPrice.model

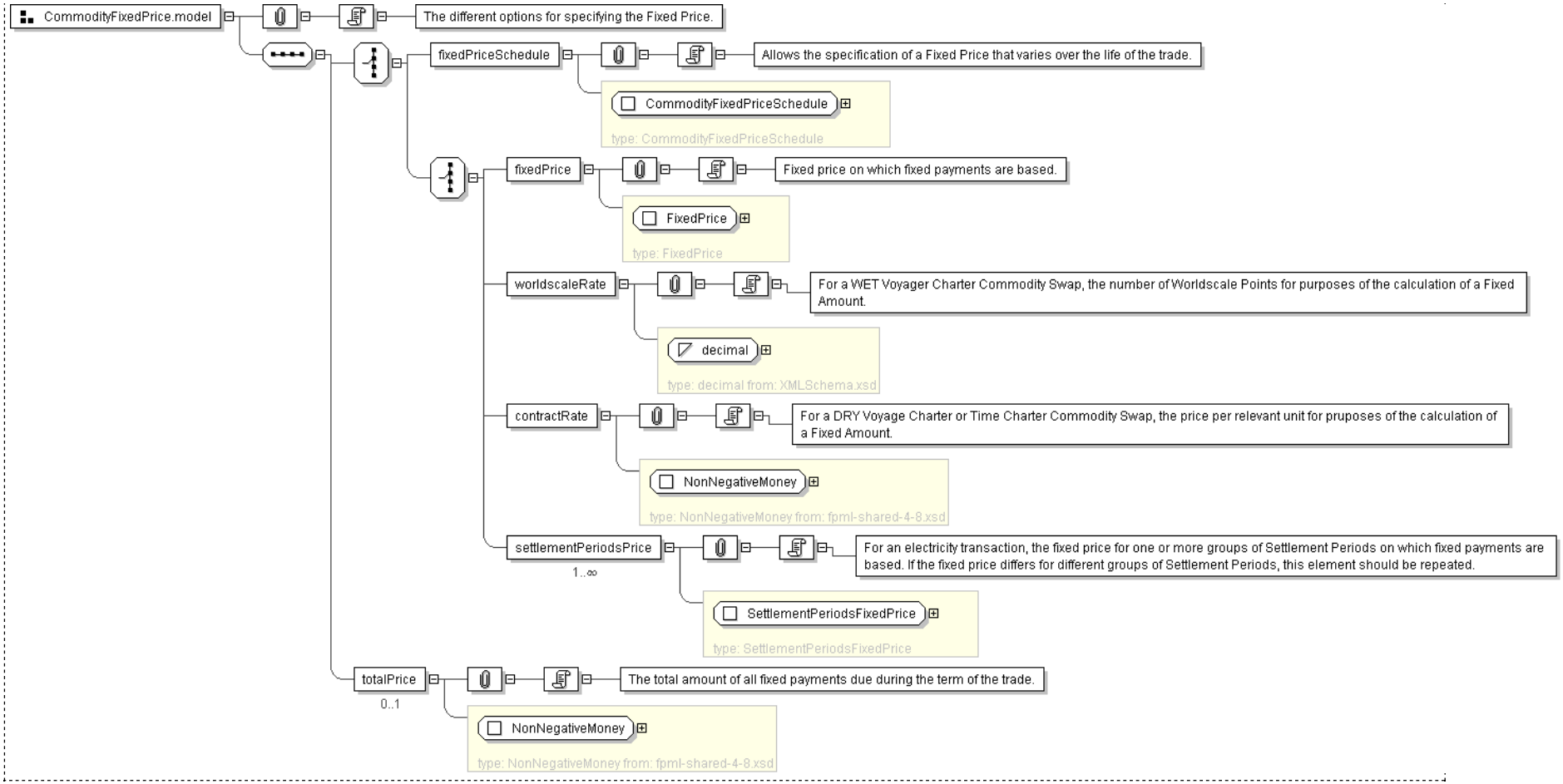
[Table of contents]

Name	CommodityFixedPrice.model
Used by (from the same schema document)	Complex Type FixedPriceLeg
Documentation	The different options for specifying the Fixed Price.

XML Instance Representation

```
Start Choice [1]
   <fixedPriceSchedule> CommodityFixedPriceSchedule </fixedPriceSchedule> [1]
   'Allows the specification of a Fixed Price that varies over the life of the trade.'

   Start Choice [1]
      <fixedPrice> FixedPrice </fixedPrice> [1]
      'Fixed price on which fixed payments are based.'

      <worldscaleRate> xsd:decimal </worldscaleRate> [1]
      'For a WET Voyager Charter Commodity Swap, the number of Worldscale Points for purposes of the calculation of a Fixed Amount.'

      <contractRate> NonNegativeMoney </contractRate> [1]
      'For a DRY Voyage Charter or Time Charter Commodity Swap, the price per relevant unit for pruposes of the calculation of a Fixed Amount.'

      <settlementPeriodsPrice> SettlementPeriodsFixedPrice </settlementPeriodsPrice> [1..*]
      'For an electricity transaction, the fixed price for one or more groups of Settlement Periods on which fixed payments are based. If the fixed price differs for
      different groups of Settlement Periods, this element should be repeated.'

   End Choice
End Choice
<totalPrice> NonNegativeMoney </totalPrice> [0..1]
'The total amount of all fixed payments due during the term of the trade.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityFixedPrice.model">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="fixedPriceSchedule" type=" CommodityFixedPriceSchedule "/>
      <xsd:choice>
        <xsd:element name="fixedPrice" type=" FixedPrice "/>
        <xsd:element name="worldscaleRate" type=" xsd:decimal "/>
        <xsd:element name="contractRate" type=" NonNegativeMoney "/>
        <xsd:element name="settlementPeriodsPrice" type=" SettlementPeriodsFixedPrice " maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:choice>
    <xsd:element name="totalPrice" type=" NonNegativeMoney " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityFreightFlatRate.model

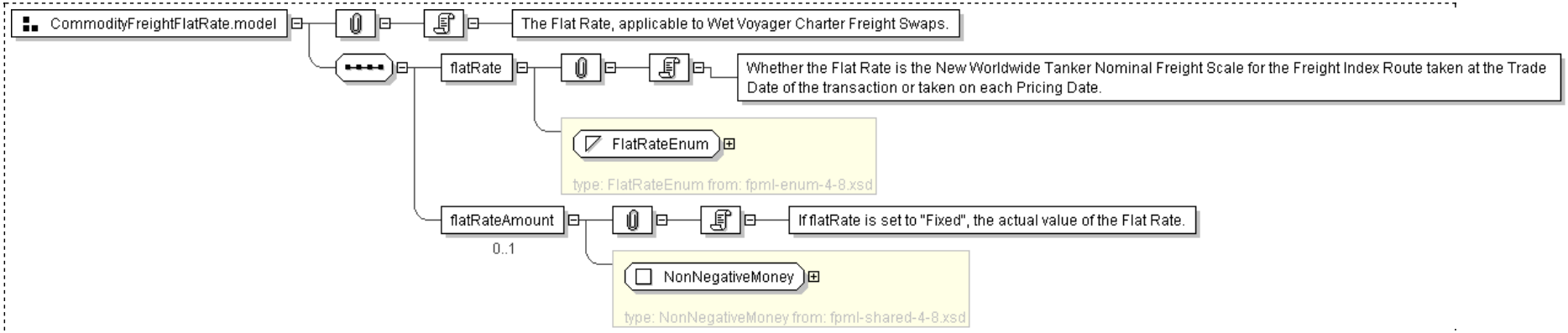
[Table of contents]

Name	CommodityFreightFlatRate.model
Used by (from the same schema document)	Complex Type FixedPriceLeg , Complex Type FloatingPriceLeg
Documentation	The Flat Rate, applicable to Wet Voyager Charter Freight Swaps.

XML Instance Representation

```
<flatRate> FlatRateEnum </flatRate> [1]
'Whether the Flat Rate is the New Worldwide Tanker Nominal Freight Scale for the Freight Index Route taken at the Trade Date of the transaction
or taken on each Pricing Date.'

<flatRateAmount> NonNegativeMoney </flatRateAmount> [0..1]
'If flatRate is set to \"Fixed\", the actual value of the Flat Rate.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityFreightFlatRate.model">
  <xsd:sequence>
    <xsd:element name="flatRate" type=" FlatRateEnum "/>
    <xsd:element name="flatRateAmount" type=" NonNegativeMoney " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityNonPeriodicPaymentDates.model

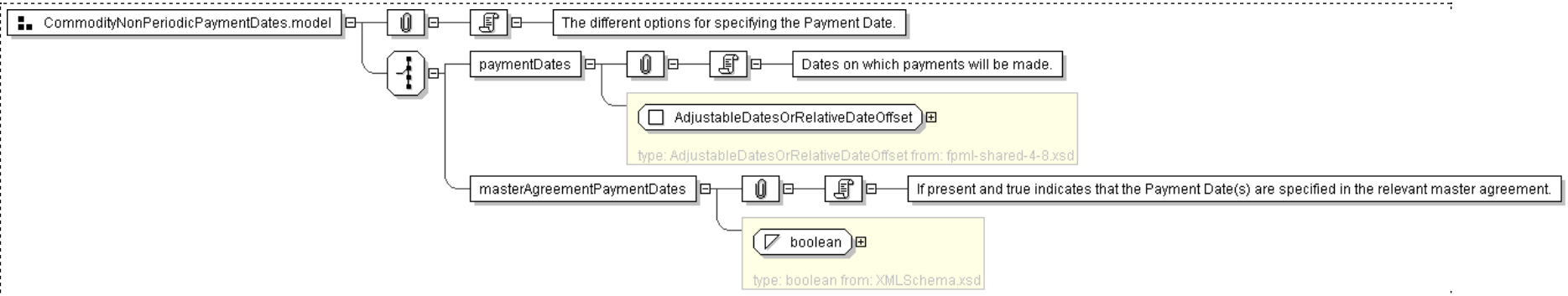
[Table of contents]

Name	CommodityNonPeriodicPaymentDates.model
Used by (from the same schema document)	Model Group CommodityPaymentDates.model
Documentation	The different options for specifying the Payment Date.

XML Instance Representation

```
Start Choice [1]
   <paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]
   'Dates on which payments will be made.'

   <masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]
   'If present and true indicates that the Payment Date(s) are specified in the relevant master agreement.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityNonPeriodicPaymentDates.model">
  <xsd:choice>
    <xsd:element name="paymentDates" type=" AdjustableDatesOrRelativeDateOffset "/>
    <xsd:element name="masterAgreementPaymentDates" type=" xsd:boolean "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityNotionalQuantity.model

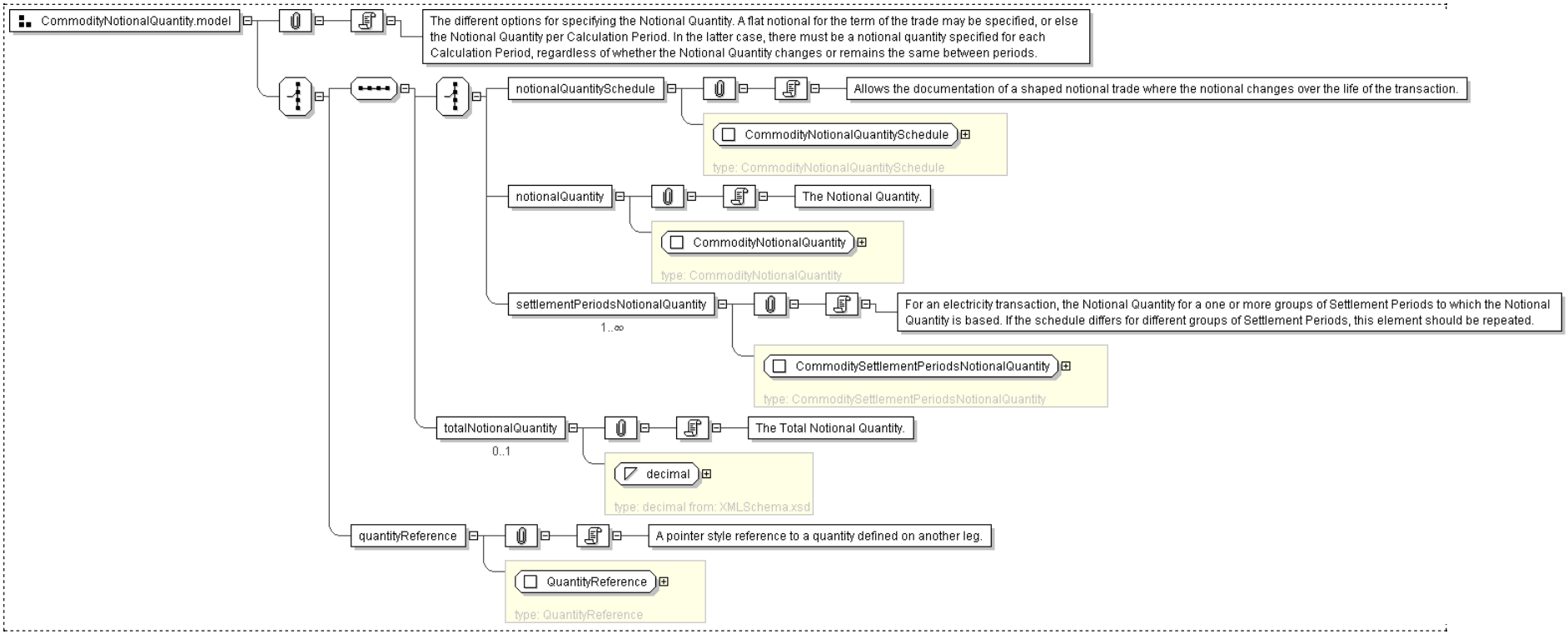
[Table of contents]

Name	CommodityNotionalQuantity.model
Used by (from the same schema document)	Complex Type FixedPriceLeg , Complex Type FloatingPriceLeg , Model Group CommodityFinancialOption.model
Documentation	The different options for specifying the Notional Quantity. A flat notional for the term of the trade may be specified, or else the Notional Quantity per Calculation Period. In the latter case, there must be a notional quantity specified for each Calculation Period, regardless of whether the Notional Quantity changes or remains the same between periods.

XML Instance Representation

```
Start Choice [1]
  Start Choice [1]
    <notionalQuantitySchedule> CommodityNotionalQuantitySchedule </notionalQuantitySchedule> [1]
    'Allows the documentation of a shaped notional trade where the notional changes over the life of the transaction.'

    <notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]
    'The Notional Quantity.'

    <settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity </settlementPeriodsNotionalQuantity> [1..*]
    'For an electricity transaction, the Notional Quantity for a one or more groups of Settlement Periods to which the Notional Quantity is based. If the schedule differs for
    different groups of Settlement Periods, this element should be repeated.'

  End Choice
  <totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]
  'The Total Notional Quantity.'

  <quantityReference> QuantityReference </quantityReference> [1]
  'A pointer style reference to a quantity defined on another leg.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityNotionalQuantity.model">
  <xsd:choice>
```

```
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="notionalQuantitySchedule" type=" CommodityNotionalQuantitySchedule "/>
        <xsd:element name="notionalQuantity" type=" CommodityNotionalQuantity "/>
        <xsd:element name="settlementPeriodsNotionalQuantity" type=" CommoditySettlementPeriodsNotionalQuantity " maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:element name="totalNotionalQuantity" type=" xsd:decimal " minOccurs="0"/>
    </xsd:sequence>
    <xsd:element name="quantityReference" type=" QuantityReference "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityPaymentDates.model

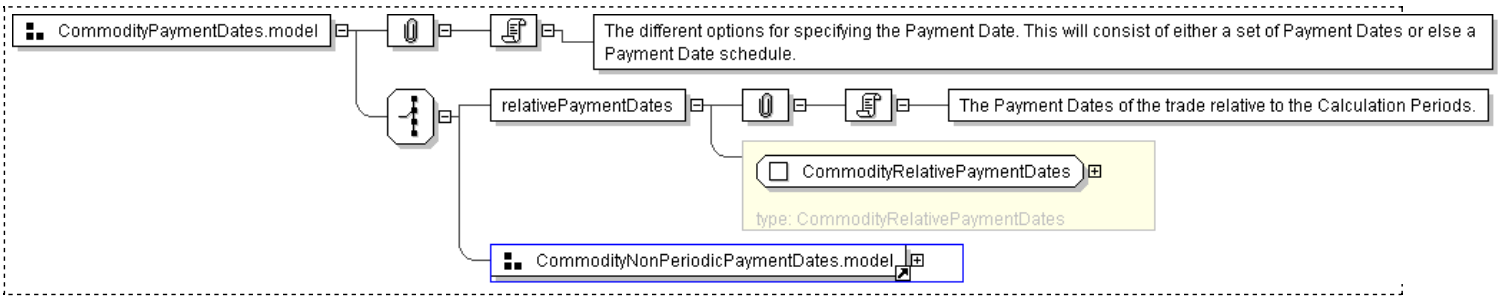
[Table of contents]

Name	CommodityPaymentDates.model
Used by (from the same schema document)	Complex Type CommodityExercise , Complex Type FixedPriceLeg , Complex Type FloatingPriceLeg , Complex Type NonPeriodicFixedPriceLeg
Documentation	The different options for specifying the Payment Date. This will consist of either a set of Payment Dates or else a Payment Date schedule.

XML Instance Representation

```
Start Choice [1]
   <relativePaymentDates> CommodityRelativePaymentDates </relativePaymentDates> [1]
   'The Payment Dates of the trade relative to the Calculation Periods.'

   Start Choice [1]
      <paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]
      'Dates on which payments will be made.'

      <masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]
      'If present and true indicates that the Payment Date(s) are specified in the relevant master agreement.'

   End Choice
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityPaymentDates.model">
   <xsd:choice>
      <xsd:element name="relativePaymentDates" type=" CommodityRelativePaymentDates "/>
      <xsd:group ref=" CommodityNonPeriodicPaymentDates.model "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityPhysicalOption.model

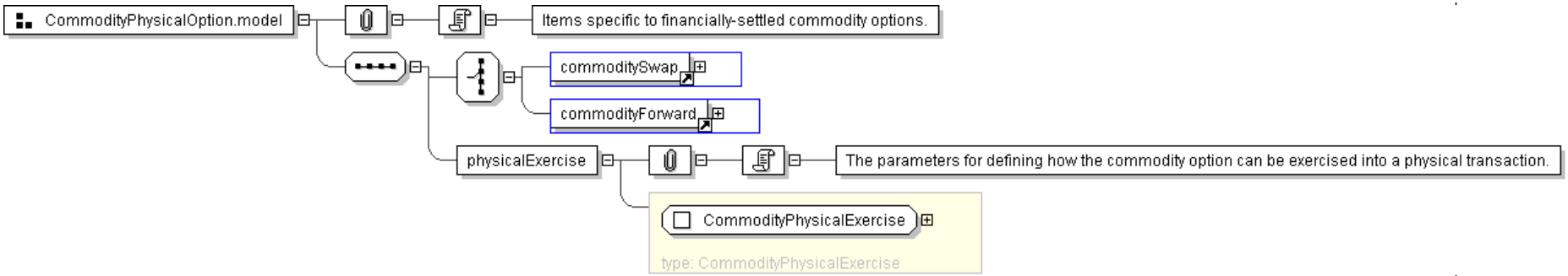
[Table of contents]

Name	CommodityPhysicalOption.model
Used by (from the same schema document)	Complex Type CommodityOption
Documentation	Items specific to financially-settled commodity options.

XML Instance Representation

```
Start Choice [1]
   <commoditySwap> ... </commoditySwap> [1]
   <commodityForward> ... </commodityForward> [1]
End Choice
<physicalExercise> CommodityPhysicalExercise </physicalExercise> [1]
'The parameters for defining how the commodity option can be exercised into a physical transaction.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityPhysicalOption.model">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element ref=" commoditySwap "/>
         <xsd:element ref=" commodityForward "/>
      </xsd:choice>
      <xsd:element name="physicalExercise" type=" CommodityPhysicalExercise "/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityStrikePrice.model

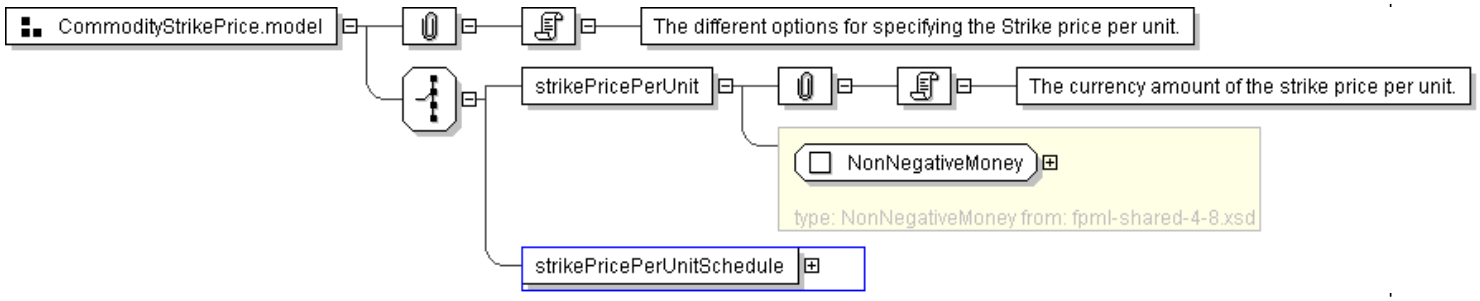
[Table of contents]

Name	CommodityStrikePrice.model
Used by (from the same schema document)	Model Group CommodityFinancialOption.model
Documentation	The different options for specifying the Strike price per unit.

XML Instance Representation

```
Start Choice [1]
   <strikePricePerUnit> NonNegativeMoney </strikePricePerUnit> [1]
   'The currency amount of the strike price per unit.'

   <strikePricePerUnitSchedule> CommodityStrikeSchedule </strikePricePerUnitSchedule> [1]
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityStrikePrice.model">
   <xsd:choice>
      <xsd:element name="strikePricePerUnit" type=" NonNegativeMoney "/>
      <xsd:element name="strikePricePerUnitSchedule" type=" CommodityStrikeSchedule "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityUSCoalDelivery.model

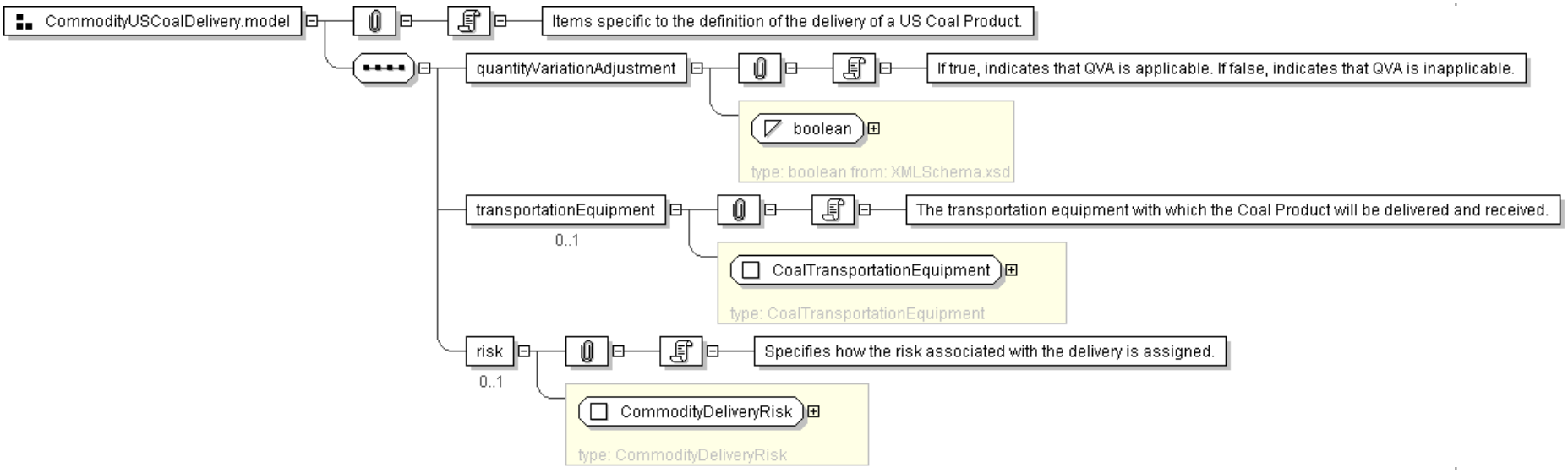
[Table of contents]

Name	CommodityUSCoalDelivery.model
Used by (from the same schema document)	Complex Type CoalDelivery
Documentation	Items specific to the definition of the delivery of a US Coal Product.

XML Instance Representation

```
<quantityVariationAdjustment> xsd:boolean </quantityVariationAdjustment> [1]
'If true, indicates that QVA is applicable. If false, indicates that QVA is inapplicable.'

<transportationEquipment> CoalTransportationEquipment </transportationEquipment> [0..1]
'The transportation equipment with which the Coal Product will be delivered and received.'

<risk> CommodityDeliveryRisk </risk> [0..1]
'Specifies how the risk associated with the delivery is assigned.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityUSCoalDelivery.model">
  <xsd:sequence>
    <xsd:element name="quantityVariationAdjustment" type=" xsd:boolean "/>
    <xsd:element name="transportationEquipment" type=" CoalTransportationEquipment " minOccurs="0"/>
    <xsd:element name="risk" type=" CommodityDeliveryRisk " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CommodityUSCoalProduct.model

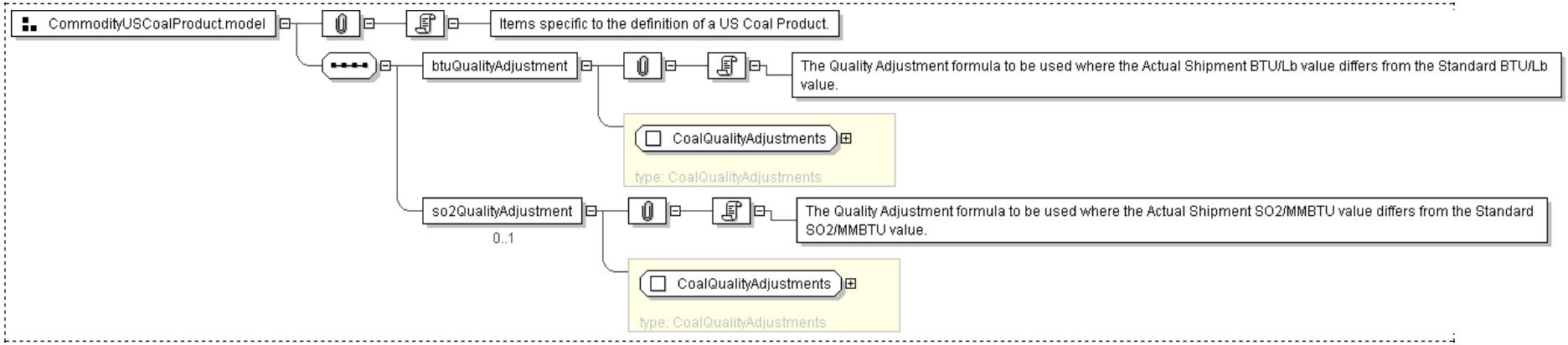
[Table of contents]

Name	CommodityUSCoalProduct.model
Used by (from the same schema document)	Complex Type CoalProduct
Documentation	Items specific to the definition of a US Coal Product.

XML Instance Representation

```
<btuQualityAdjustment> CoalQualityAdjustments </btuQualityAdjustment> [1]
'The Quality Adjustment formula to be used where the Actual Shipment BTU/Lb value differs from the Standard BTU/Lb value.'

<so2QualityAdjustment> CoalQualityAdjustments </so2QualityAdjustment> [0..1]
'The Quality Adjustment formula to be used where the Actual Shipment SO2/MMBTU value differs from the Standard SO2/MMBTU value.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CommodityUSCoalProduct.model">
  <xsd:sequence>
    <xsd:element name="btuQualityAdjustment" type=" CoalQualityAdjustments "/>
    <xsd:element name="so2QualityAdjustment" type=" CoalQualityAdjustments " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: LagOrReference.model

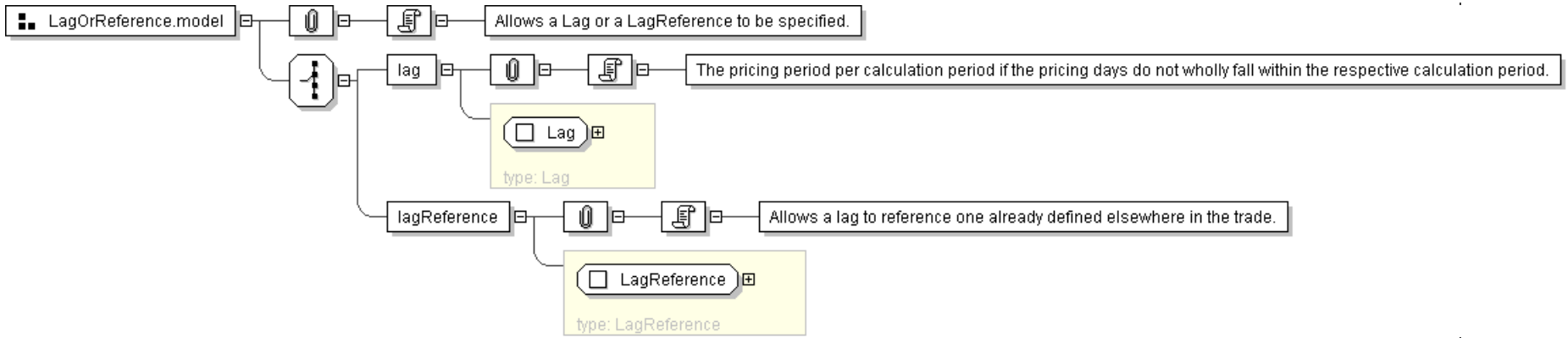
[Table of contents]

Name	LagOrReference.model
Used by (from the same schema document)	Complex Type CommodityFx
Documentation	Allows a Lag or a LagReference to be specified.

XML Instance Representation

```
Start Choice [1]
   <lag> Lag </lag> [1]
   'The pricing period per calculation period if the pricing days do not wholly fall within the respective calculation period.'

   <lagReference> LagReference </lagReference> [1]
   'Allows a lag to reference one already defined elsewhere in the trade.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="LagOrReference.model">
   <xsd:choice>
      <xsd:element name="lag" type=" Lag "/>
      <xsd:element name="lagReference" type=" LagReference "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Price.model

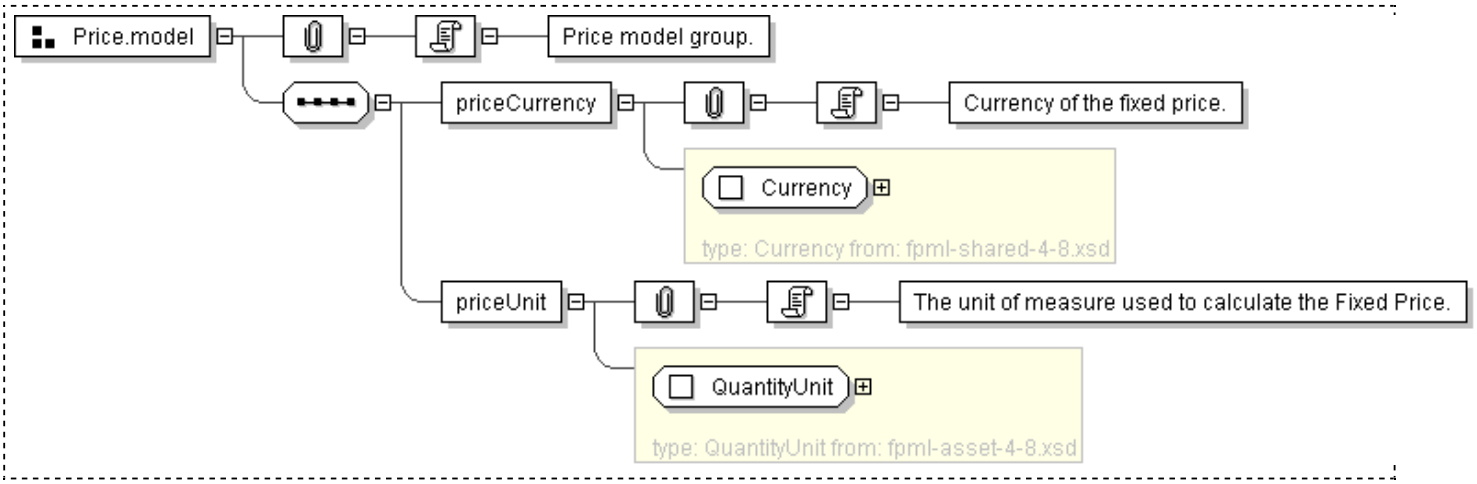
[Table of contents]

Name	Price.model
Used by (from the same schema document)	Complex Type FixedPrice
Documentation	Price model group.

XML Instance Representation

```
<priceCurrency> Currency </priceCurrency> [1]
'Currency of the fixed price.'

<priceUnit> QuantityUnit </priceUnit> [1]
'The unit of measure used to calculate the Fixed Price.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Price.model">
  <xsd:sequence>
    <xsd:element name="priceCurrency" type=" Currency "/>
    <xsd:element name="priceUnit" type=" QuantityUnit "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PricingDays.model

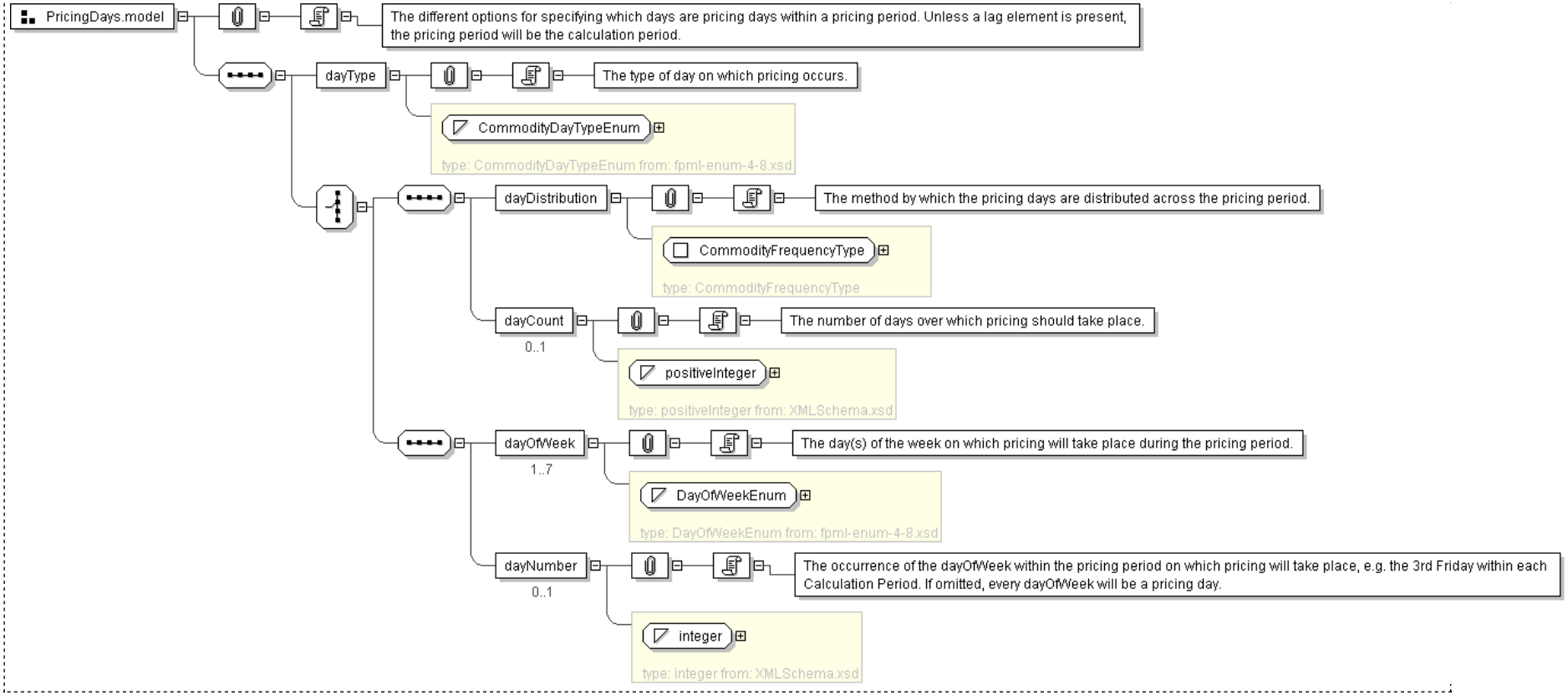
[Table of contents]

Name	PricingDays.model
Used by (from the same schema document)	Complex Type CommodityFx , Complex Type CommodityPricingDates
Documentation	The different options for specifying which days are pricing days within a pricing period. Unless a lag element is present, the pricing period will be the calculation period.

XML Instance Representation

```
<dayType> CommodityDayTypeEnum </dayType> [1]
'The type of day on which pricing occurs.'

Start Choice [1]
   <dayDistribution> CommodityFrequencyType </dayDistribution> [1]
   'The method by which the pricing days are distributed across the pricing period.'

   <dayCount> xsd:positiveInteger </dayCount> [0..1]
   'The number of days over which pricing should take place.'

   <dayOfWeek> DayOfWeekEnum </dayOfWeek> [1..7]
   'The day(s) of the week on which pricing will take place during the pricing period.'

   <dayNumber> xsd:integer </dayNumber> [0..1]
   'The occurrence of the dayOfWeek within the pricing period on which pricing will take place, e.g. the 3rd Friday within each Calculation Period. If
   omitted, every dayOfWeek will be a pricing day.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="PricingDays.model">
  <xsd:sequence>
    <xsd:element name="dayType" type=" CommodityDayTypeEnum "/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="dayDistribution" type=" CommodityFrequencyType "/>
        <xsd:element name="dayCount" type=" xsd:positiveInteger " minOccurs="0"/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="dayOfWeek" type=" DayOfWeekEnum " maxOccurs="7"/>
        <xsd:element name="dayNumber" type=" xsd:integer " minOccurs="0"/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AbsoluteTolerance

[Table of contents]

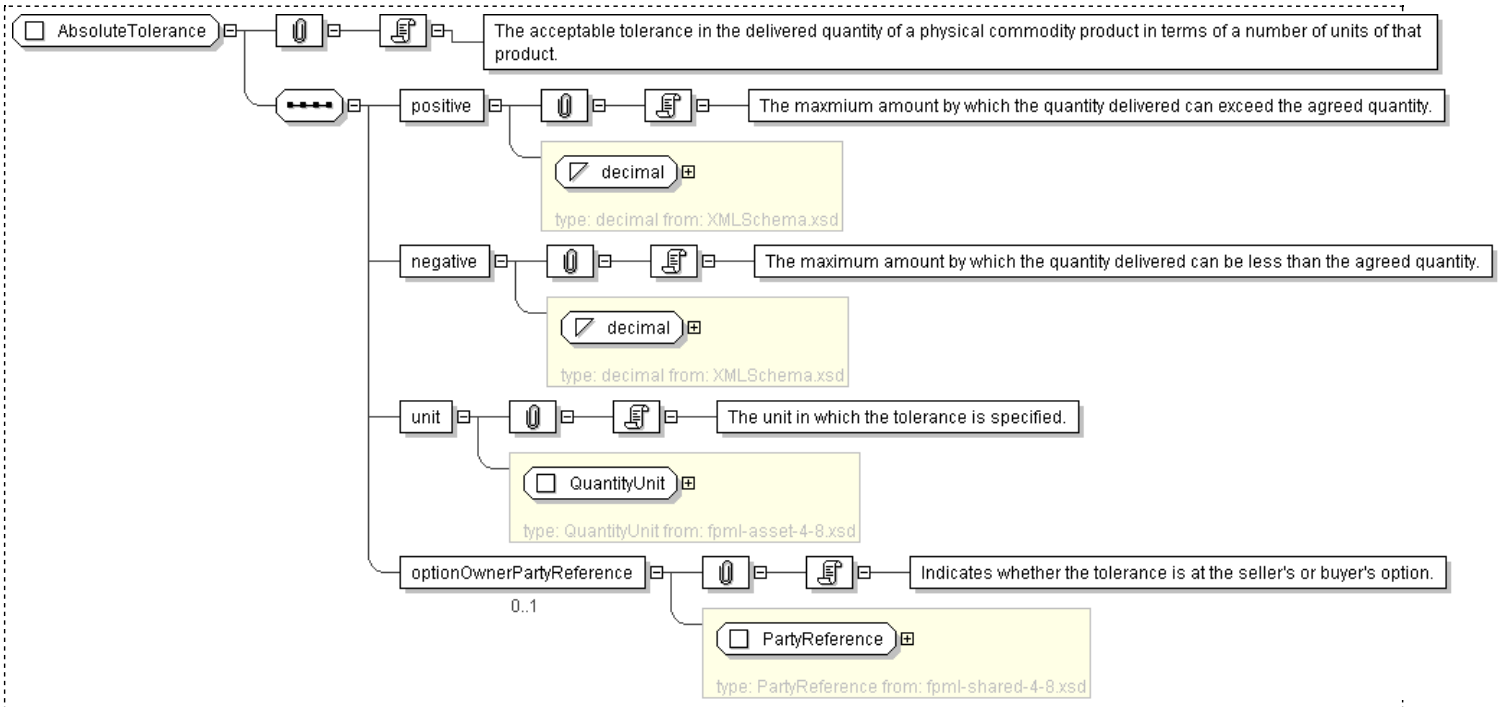
Super-types:	None
Sub-types:	None

Name	AbsoluteTolerance
Used by (from the same schema document)	Complex Type OilDelivery
Abstract	no
Documentation	The acceptable tolerance in the delivered quantity of a physical commodity product in terms of a number of units of that product.

XML Instance Representation

```
<...>
  <positive> xsd:decimal </positive> [1]
  'The maxmium amount by which the quantity delivered can exceed the agreed quantity.'

  <negative> xsd:decimal </negative> [1]
  'The maximum amount by which the quantity delivered can be less than the agreed quantity.'

  <unit> QuantityUnit </unit> [1]
  'The unit in which the tolerance is specified.'

  <optionOwnerPartyReference> PartyReference </optionOwnerPartyReference> [0..1]
  'Indicates whether the tolerance is at the seller\'s or buyer\'s option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AbsoluteTolerance">
  <xsd:sequence>
    <xsd:element name="positive" type=" xsd:decimal "/>
    <xsd:element name="negative" type=" xsd:decimal "/>
    <xsd:element name="unit" type=" QuantityUnit "/>
    <xsd:element name="optionOwnerPartyReference" type=" PartyReference " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BullionDeliveryLocation

[Table of contents]

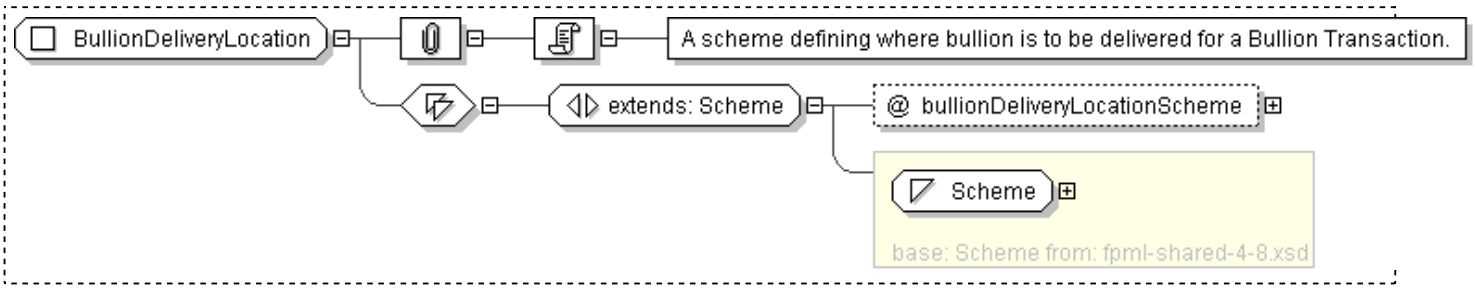
Super-types:	Scheme < BullionDeliveryLocation (by extension)
Sub-types:	None

Name	BullionDeliveryLocation
Used by (from the same schema document)	Complex Type BullionPhysicalLeg
Abstract	no
Documentation	A scheme defining where bullion is to be delivered for a Bullion Transaction.

XML Instance Representation

```
<...
 bullionDeliveryLocationScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BullionDeliveryLocation">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="bullionDeliveryLocationScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/bullion-delivery-location"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BullionPhysicalLeg

[Table of contents]

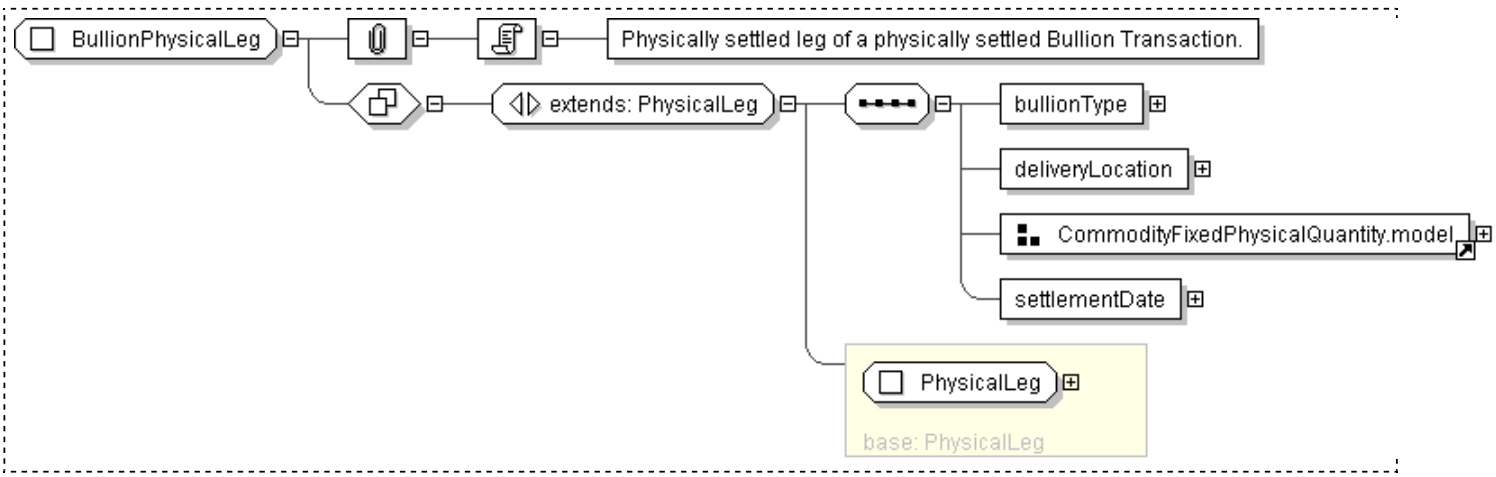
Super-types:	Leg < PhysicalLeg (by extension) < BullionPhysicalLeg (by extension)
Sub-types:	None

Name	BullionPhysicalLeg
Used by (from the same schema document)	Complex Type CommodityForward
Abstract	no
Documentation	Physically settled leg of a physically settled Bullion Transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <bullionType> BullionTypeEnum </bullionType> [1]
   'The type of Bullion underlying a Bullion Transaction.'

   <deliveryLocation> BullionDeliveryLocation </deliveryLocation> [1]
   'The physical delivery location for the transaction.'

   Start Choice [1]
      <physicalQuantity> CommodityNotionalQuantity </physicalQuantity> [1]
      'The Quantity per Delivery Period.'

      <physicalQuantitySchedule> CommodityPhysicalQuantitySchedule </physicalQuantitySchedule> [1]
      'Allows the documentation of a shaped quantity trade where the quantity changes over
      the life of the transaction.'

   End Choice
   <totalPhysicalQuantity> UnitQuantity </totalPhysicalQuantity> [0..1]
   'The Total Quantity of the commodity to be delivered.'

   <settlementDate> AdjustableOrRelativeDate </settlementDate> [1]
   'Date on which the bullion will settle.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BullionPhysicalLeg">
  <xsd:complexContent>
    <xsd:extension base=" PhysicalLeg ">
      <xsd:sequence>
        <xsd:element name="bullionType" type=" BullionTypeEnum "/>
        <xsd:element name="deliveryLocation" type=" BullionDeliveryLocation "/>
        <xsd:group ref=" CommodityFixedPhysicalQuantity.model "/>
        <xsd:element name="settlementDate" type=" AdjustableOrRelativeDate "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodsDatesReference

[Table of contents]

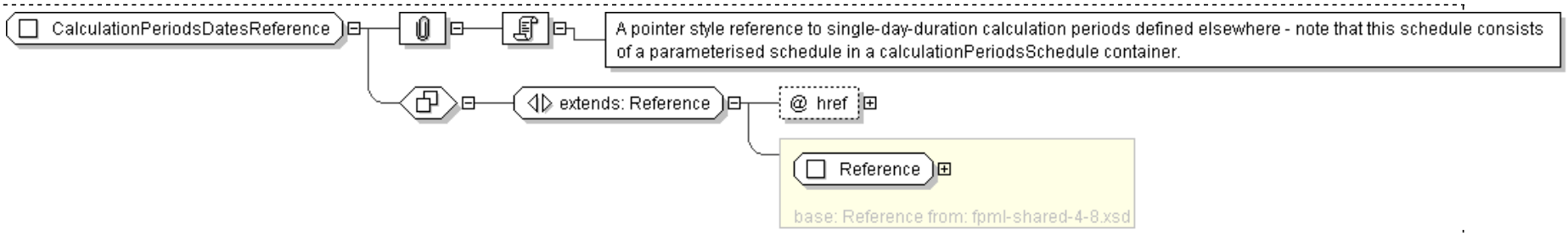
Super-types:	Reference < CalculationPeriodsDatesReference (by extension)
Sub-types:	None

Name	CalculationPeriodsDatesReference
Used by (from the same schema document)	Model Group CommodityCalculationPeriodsPointer.model
Abstract	no
Documentation	A pointer style reference to single-day-duration calculation periods defined elsewhere - note that this schedule consists of a parameterised schedule in a calculationPeriodsSchedule container.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodsDatesReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="CommodityCalculationPeriodsSchedule"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodsReference

[Table of contents]

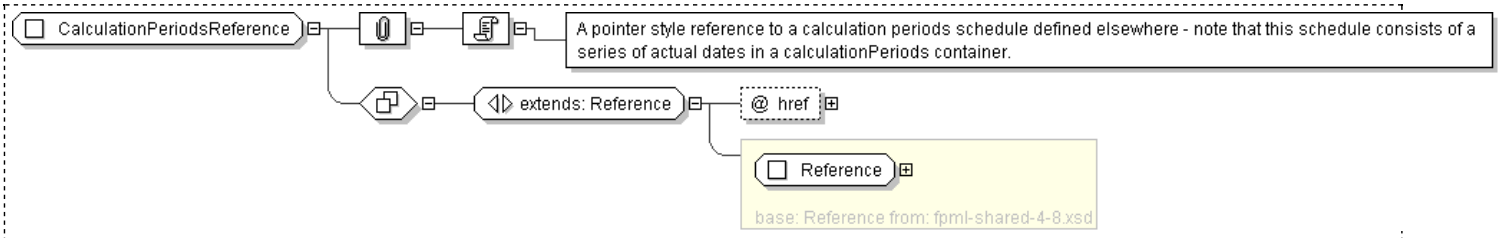
Super-types:	Reference < CalculationPeriodsReference (by extension)
Sub-types:	None

Name	CalculationPeriodsReference
Used by (from the same schema document)	Model Group CommodityDeliveryPeriodsPointer.model , Model Group CommodityCalculationPeriodsPointer.model
Abstract	no
Documentation	A pointer style reference to a calculation periods schedule defined elsewhere - note that this schedule consists of a series of actual dates in a calculationPeriods container.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodsReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="AdjustableDates"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodsScheduleReference

[Table of contents]

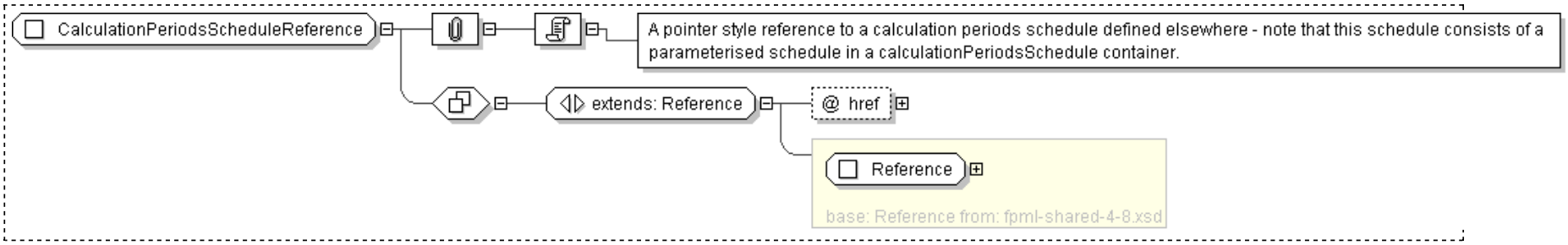
Super-types:	Reference < CalculationPeriodsScheduleReference (by extension)
Sub-types:	None

Name	CalculationPeriodsScheduleReference
Used by (from the same schema document)	Model Group CommodityDeliveryPeriodsPointer.model , Model Group CommodityCalculationPeriodsPointer.model
Abstract	no
Documentation	A pointer style reference to a calculation periods schedule defined elsewhere - note that this schedule consists of a parameterised schedule in a calculationPeriodsSchedule container.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodsScheduleReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="CommodityCalculationPeriodsSchedule"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalAttributeDecimal

[Table of contents]

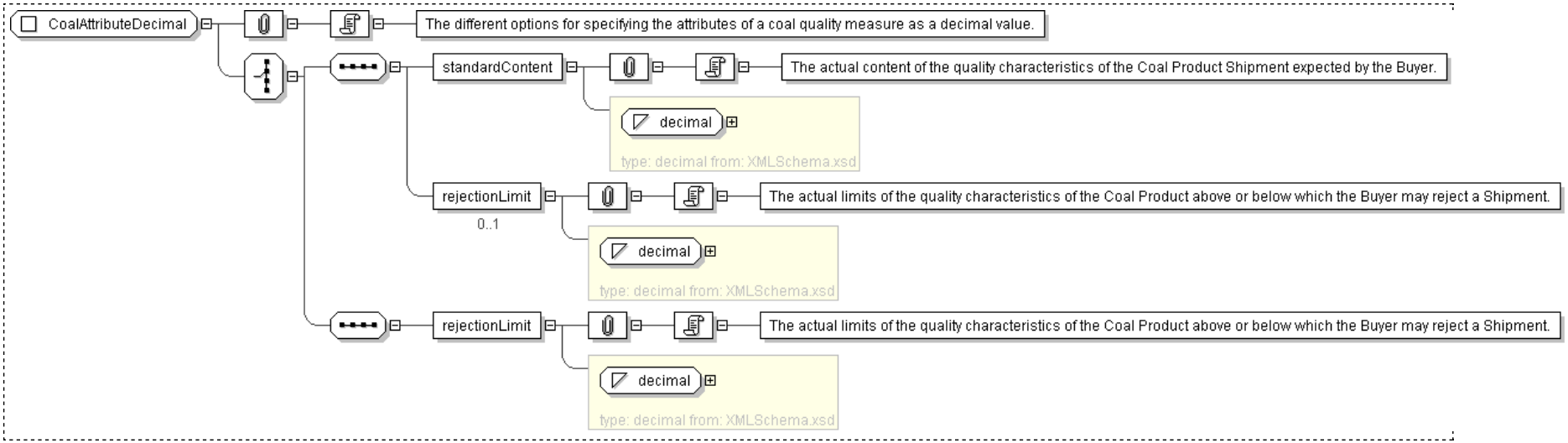
Super-types:	None
Sub-types:	None

Name	CoalAttributeDecimal
Used by (from the same schema document)	Model Group CommodityCoalProperties.model , Model Group CommodityCoalProperties.model , Model Group CommodityCoalProperties.model , Model Group CommodityCoalProperties.model , Model Group CommodityCoalProperties.model , Model Group CommodityCoalReducingAtmosphere.model , Model Group CommodityCoalReducingAtmosphere.model , Model Group CommodityCoalReducingAtmosphere.model , Model Group CommodityCoalReducingAtmosphere.model
Abstract	no
Documentation	The different options for specifying the attributes of a coal quality measure as a decimal value.

XML Instance Representation

```
<...>
  Start Choice [1]
    <standardContent> xsd:decimal </standardContent> [1]
     'The actual content of the quality characteristics of the Coal Product Shipment expected by the Buyer.'

    <rejectionLimit> xsd:decimal </rejectionLimit> [0..1]
     'The actual limits of the quality characteristics of the Coal Product above or below which the Buyer may reject a Shipment.'

    <rejectionLimit> xsd:decimal </rejectionLimit> [1]
     'The actual limits of the quality characteristics of the Coal Product above or below which the Buyer may reject a Shipment.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalAttributeDecimal">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="standardContent" type=" xsd:decimal "/>
```

```
        <xsd:element name="rejectionLimit" type=" xsd:decimal " minOccurs="0"/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="rejectionLimit" type=" xsd:decimal "/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalAttributePercentage

[Table of contents]

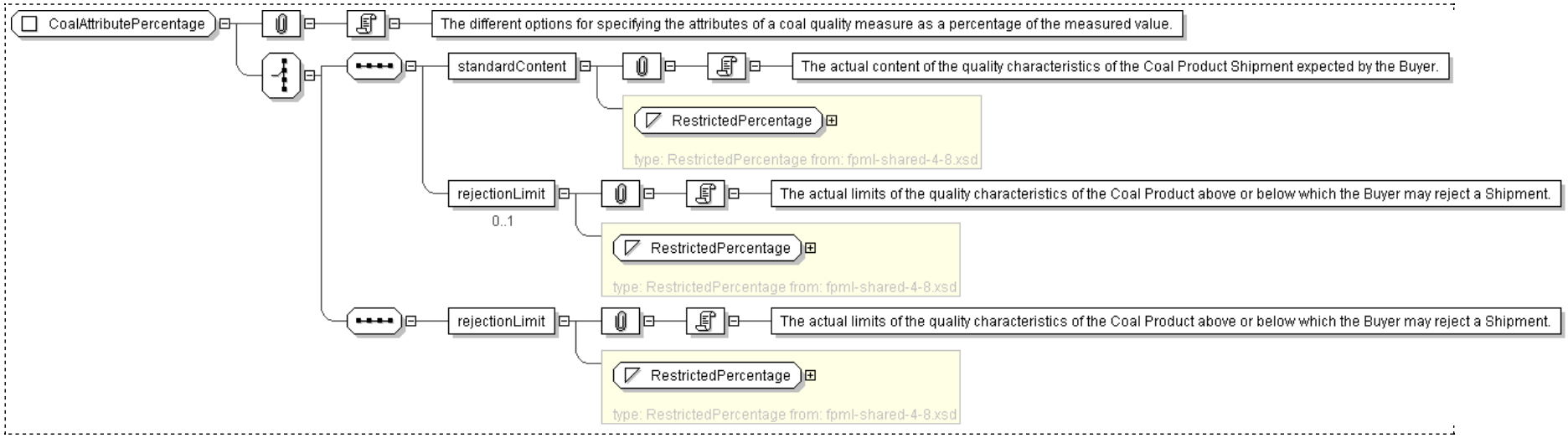
Super-types:	None
Sub-types:	None

Name	CoalAttributePercentage
Used by (from the same schema document)	Model Group CommodityCoalComposition.model , Model Group CommodityCoalComposition.model , Model Group CommodityCoalComposition.model , Model Group CommodityCoalComposition.model , Model Group CommodityCoalComposition.model
Abstract	no
Documentation	The different options for specifying the attributes of a coal quality measure as a percentage of the measured value.

XML Instance Representation

```
<...>
  Start Choice [1]
    <standardContent> RestrictedPercentage </standardContent> [1]
     'The actual content of the quality characteristics of the Coal Product Shipment expected by the Buyer.'

    <rejectionLimit> RestrictedPercentage </rejectionLimit> [0..1]
     'The actual limits of the quality characteristics of the Coal Product above or below which the Buyer may reject a Shipment.'

    <rejectionLimit> RestrictedPercentage </rejectionLimit> [1]
     'The actual limits of the quality characteristics of the Coal Product above or below which the Buyer may reject a Shipment.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalAttributePercentage">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="standardContent" type=" RestrictedPercentage "/>
      <xsd:element name="rejectionLimit" type=" RestrictedPercentage " minOccurs="0"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="              " type=" ____________________ "/>
```

```
                        rejectionLimit          RestrictedPercentage
      </xsd:sequence>
    </xsd:choice>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalDelivery

[Table of contents]

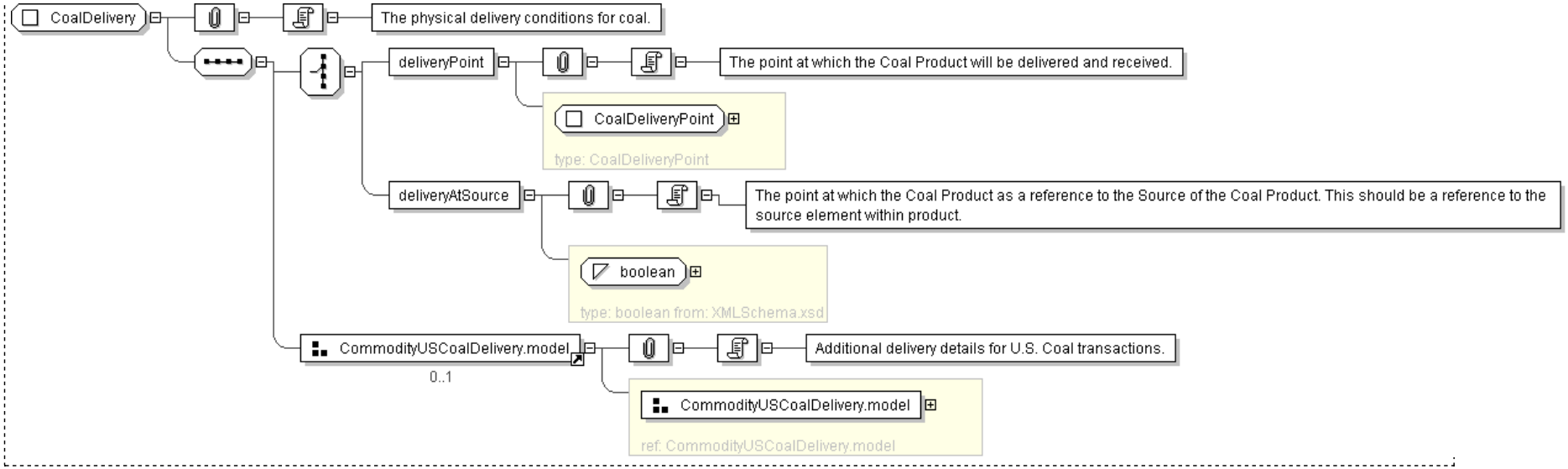
Super-types:	None
Sub-types:	None

Name	CoalDelivery
Used by (from the same schema document)	Complex Type CoalPhysicalLeg
Abstract	no
Documentation	The physical delivery conditions for coal.

XML Instance Representation

```
<...>
  Start Choice [1]
    <deliveryPoint> CoalDeliveryPoint </deliveryPoint> [1]
    'The point at which the Coal Product will be delivered and received.'

    <deliveryAtSource> xsd:boolean </deliveryAtSource> [1]
    'The point at which the Coal Product as a reference to the Source of the Coal Product. This should be a reference to the source element
    within product.'

  End Choice
  Start Group: CommodityUSCoalDelivery.model [0..1]
'Additional delivery details for U.S. Coal transactions.'


    <quantityVariationAdjustment> xsd:boolean </quantityVariationAdjustment> [1]
    'If true, indicates that QVA is applicable. If false, indicates that QVA is inapplicable.'

    <transportationEquipment> CoalTransportationEquipment </transportationEquipment> [0..1]
    'The transportation equipment with which the Coal Product will be delivered and received.'

    <risk> CommodityDeliveryRisk </risk> [0..1]
    'Specifies how the risk associated with the delivery is assigned.'

  End Group: CommodityUSCoalDelivery.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalDelivery">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="deliveryPoint" type=" CoalDeliveryPoint "/>
      <xsd:element name="deliveryAtSource" type=" xsd:boolean "/>
    </xsd:choice>
    <xsd:group ref=" CommodityUSCoalDelivery.model " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalDeliveryPoint

[Table of contents]

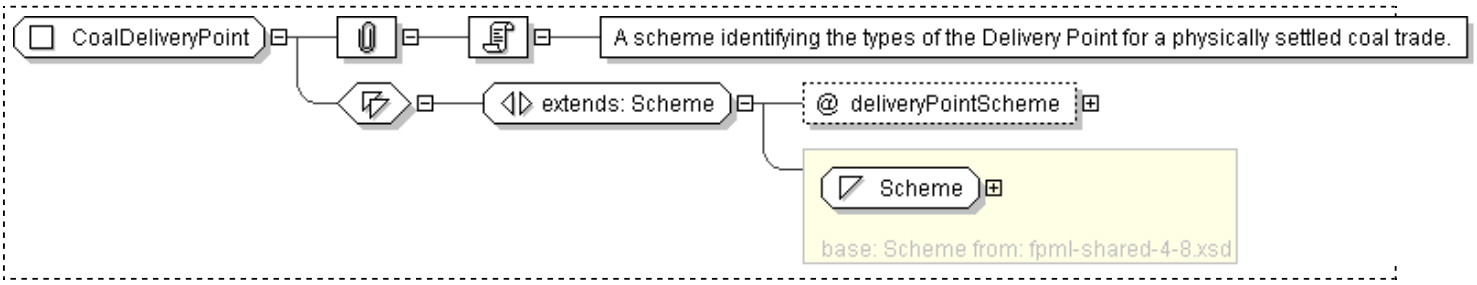
Super-types:	Scheme < CoalDeliveryPoint (by extension)
Sub-types:	None

Name	CoalDeliveryPoint
Used by (from the same schema document)	Complex Type CoalDelivery
Abstract	no
Documentation	A scheme identifying the types of the Delivery Point for a physically settled coal trade.

XML Instance Representation

```
<...
 deliveryPointScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalDeliveryPoint">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="deliveryPointScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalPhysicalLeg

[Table of contents]

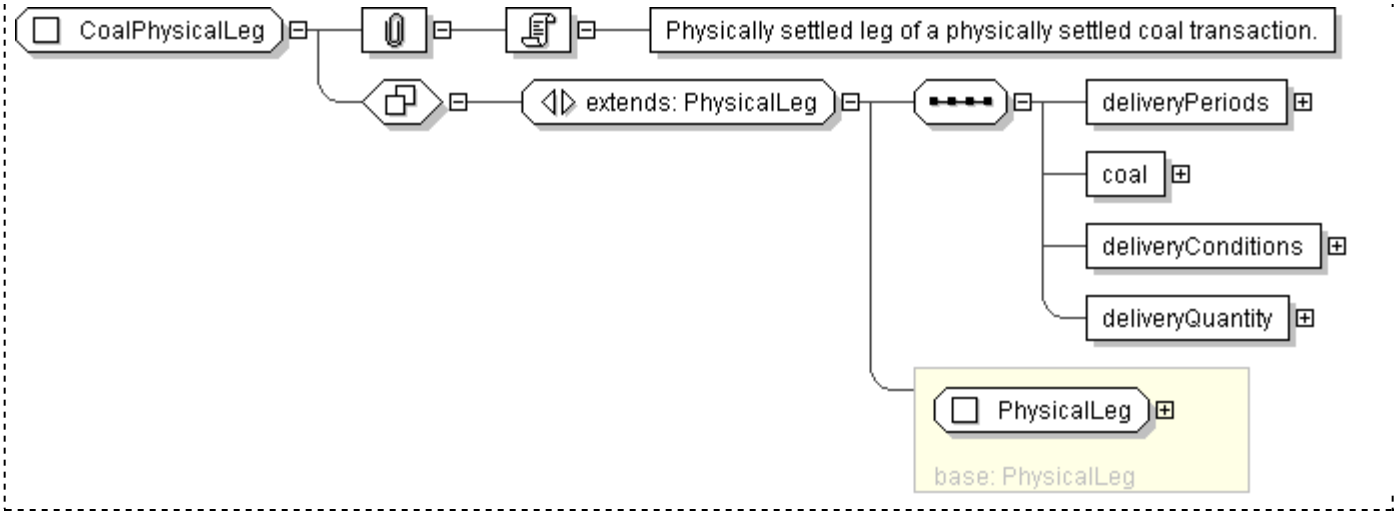
Super-types:	Leg < PhysicalLeg (by extension) < CoalPhysicalLeg (by extension)
Sub-types:	None

Name	CoalPhysicalLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Physically settled leg of a physically settled coal transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <deliveryPeriods> CommodityDeliveryPeriods </deliveryPeriods> [1]
   'The period during which delivery/deliveries of Coal Products may be scheduled.
   Equivalent to Nomination Period(s) for US Coal.'

   <coal> CoalProduct </coal> [1]
   'The specification of the Coal Product to be delivered.'

   <deliveryConditions> CoalDelivery </deliveryConditions> [1]
   'The physical delivery conditions for the transaction.'

   <deliveryQuantity> CommodityPhysicalQuantity </deliveryQuantity> [1]
   'The different options for specifying the quantity.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalPhysicalLeg">
  <xsd:complexContent>
    <xsd:extension base=" PhysicalLeg ">
      <xsd:sequence>
        <xsd:element name="deliveryPeriods" type=" CommodityDeliveryPeriods "/>
        <xsd:element name="coal" type=" CoalProduct "/>
        <xsd:element name="deliveryConditions" type=" CoalDelivery "/>
        <xsd:element name="deliveryQuantity" type=" CommodityPhysicalQuantity "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalProduct

[Table of contents]

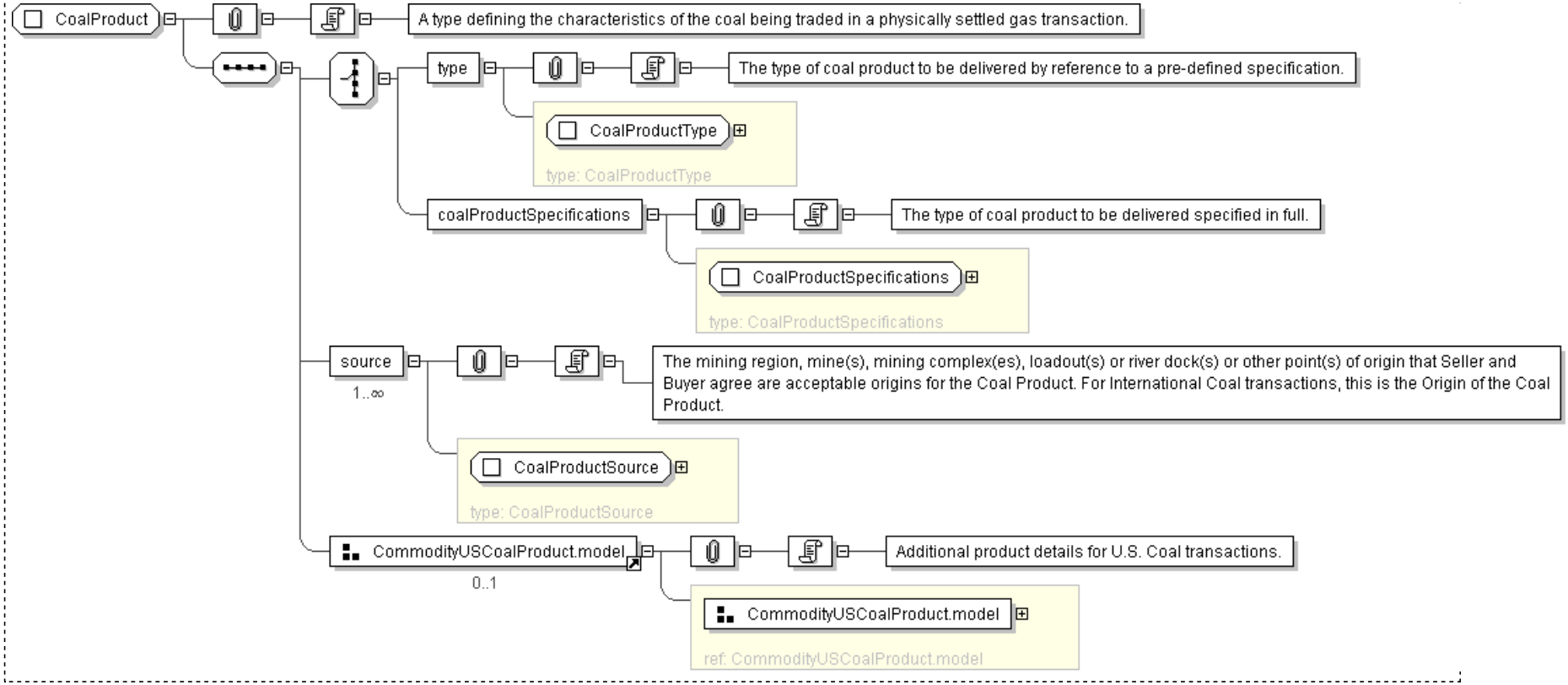
Super-types:	None
Sub-types:	None

Name	CoalProduct
Used by (from the same schema document)	Complex Type CoalPhysicalLeg
Abstract	no
Documentation	A type defining the characteristics of the coal being traded in a physically settled gas transaction.

XML Instance Representation

```
<...>
  Start Choice [1]
    <type> CoalProductType </type> [1]
    'The type of coal product to be delivered by reference to a pre-defined specification.'

    <coalProductSpecifications> CoalProductSpecifications </coalProductSpecifications> [1]
    'The type of coal product to be delivered specified in full.'

  End Choice
  <source> CoalProductSource </source> [1..*]
  'The mining region, mine(s), mining complex(es), loadout(s) or river dock(s) or other point(s) of origin that Seller and Buyer
  agree are acceptable origins for the Coal Product. For International Coal transactions, this is the Origin of the Coal
  Product.'

  Start Group: CommodityUSCoalProduct.model [0..1]
'Additional product details for U.S. Coal transactions.'


    <btuQualityAdjustment> CoalQualityAdjustments </btuQualityAdjustment> [1]
    'The Quality Adjustment formula to be used where the Actual Shipment BTU/Lb value differs from the Standard BTU/Lb value.'

    <so2QualityAdjustment> CoalQualityAdjustments </so2QualityAdjustment> [0..1]
    'The Quality Adjustment formula to be used where the Actual Shipment SO2/MMBTU value differs from the Standard SO2/MMBTU
    value.'

  End Group: CommodityUSCoalProduct.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalProduct">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="type" type=" CoalProductType "/>
      <xsd:element name="coalProductSpecifications" type=" CoalProductSpecifications "/>
    </xsd:choice>
    <xsd:element name="source" type=" CoalProductSource " maxOccurs="unbounded"/>
    <xsd:group ref=" CommodityUSCoalProduct.model " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalProductSource

[Table of contents]

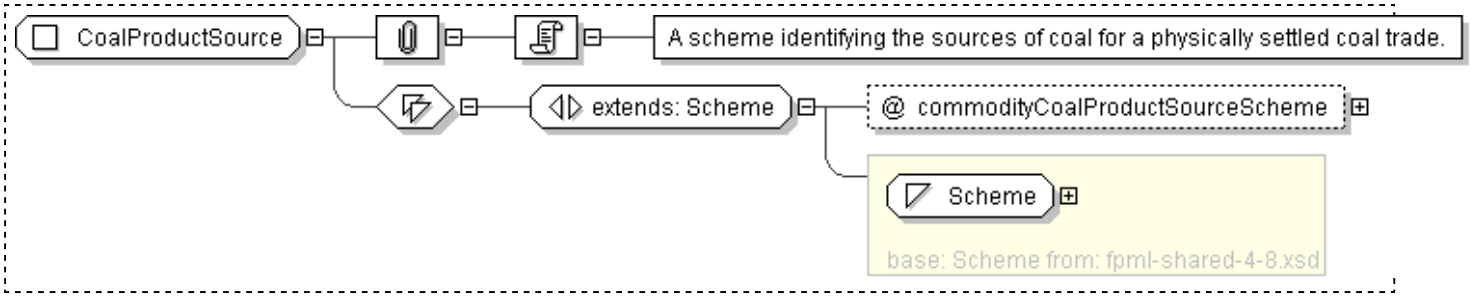
Super-types:	Scheme < CoalProductSource (by extension)
Sub-types:	None

Name	CoalProductSource
Used by (from the same schema document)	Complex Type CoalProduct
Abstract	no
Documentation	A scheme identifying the sources of coal for a physically settled coal trade.

XML Instance Representation

```
<...
 commodityCoalProductSourceScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalProductSource">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityCoalProductSourceScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-coal-product-source"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalProductSpecifications

[Table of contents]

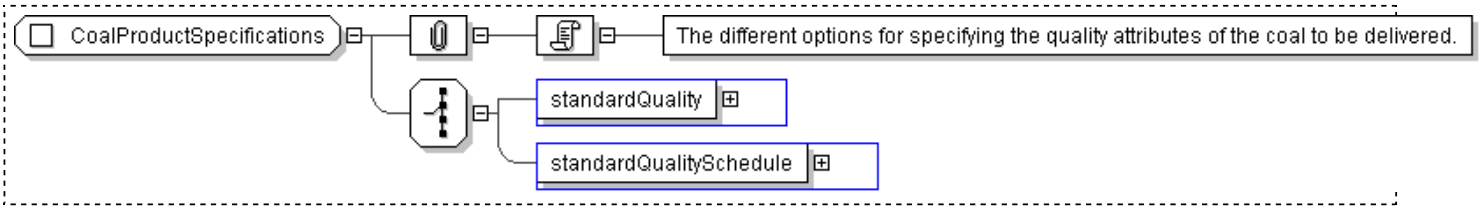
Super-types:	None
Sub-types:	None

Name	CoalProductSpecifications
Used by (from the same schema document)	Complex Type CoalProduct
Abstract	no
Documentation	The different options for specifying the quality attributes of the coal to be delivered.

XML Instance Representation

```
<...>
  Start Choice [1]
    <standardQuality> CoalStandardQuality </standardQuality> [1]
    <standardQualitySchedule> CoalStandardQualitySchedule </standardQualitySchedule> [1]
  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalProductSpecifications">
  <xsd:choice>
    <xsd:element name="standardQuality" type=" CoalStandardQuality "/>
    <xsd:element name="standardQualitySchedule" type=" CoalStandardQualitySchedule "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalProductType

[Table of contents]

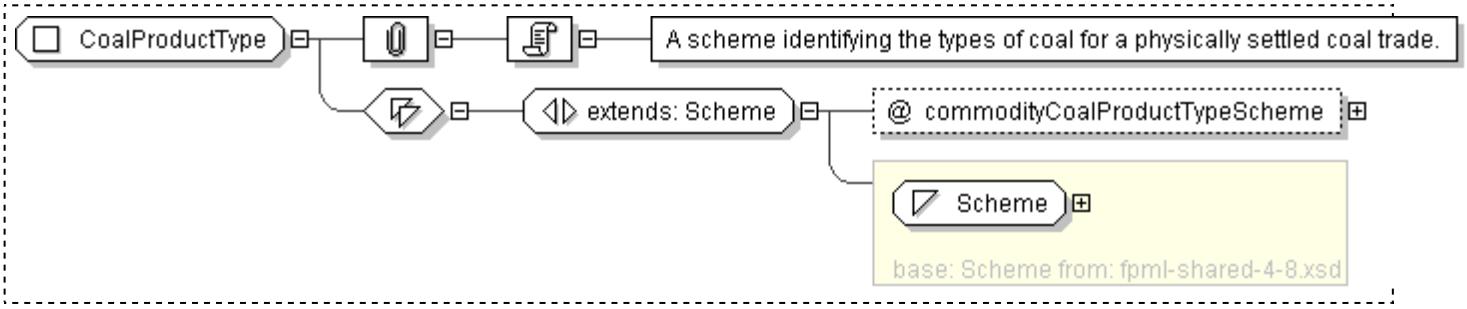
Super-types:	Scheme < CoalProductType (by extension)
Sub-types:	None

Name	CoalProductType
Used by (from the same schema document)	Complex Type CoalProduct
Abstract	no
Documentation	A scheme identifying the types of coal for a physically settled coal trade.

XML Instance Representation

```
<...
 commodityCoalProductTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalProductType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityCoalProductTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-coal-product-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalQualityAdjustments

[Table of contents]

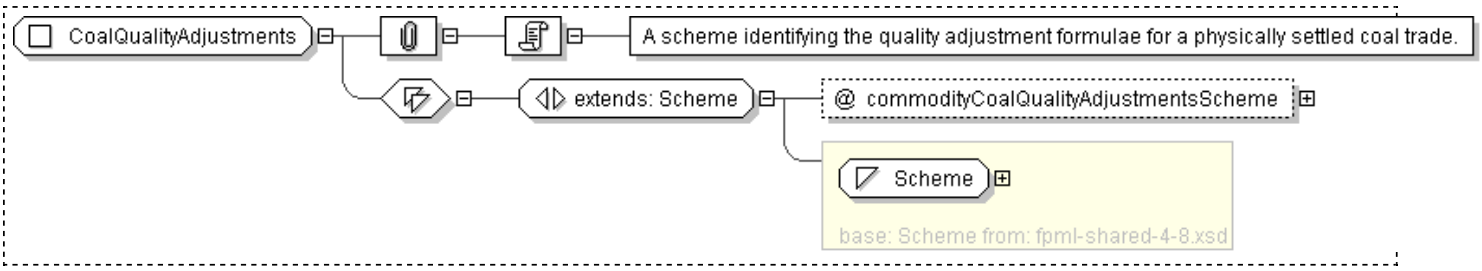
Super-types:	Scheme < CoalQualityAdjustments (by extension)
Sub-types:	None

Name	CoalQualityAdjustments
Used by (from the same schema document)	Model Group CommodityUSCoalProduct.model , Model Group CommodityUSCoalProduct.model
Abstract	no
Documentation	A scheme identifying the quality adjustment formulae for a physically settled coal trade.

XML Instance Representation

```
<...
 commodityCoalQualityAdjustmentsScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalQualityAdjustments">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="commodityCoalQualityAdjustmentsScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/commodity-coal-quality-adjustments"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalStandardQuality

[Table of contents]

Super-types:	None
Sub-types:	None

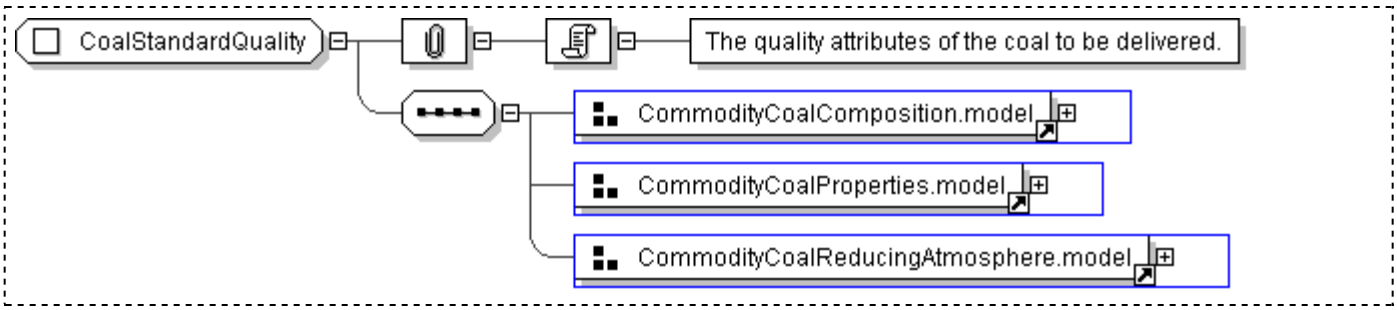
Name	CoalStandardQuality
Used by (from the same schema document)	Complex Type CoalProductSpecifications , Complex Type CoalStandardQualitySchedule
Abstract	no
Documentation	The quality attributes of the coal to be delivered.

XML Instance Representation

```
<...>
   <moisture> CoalAttributePercentage </moisture> [0..1]
   'The moisture content of the coal product.'

   <ash> CoalAttributePercentage </ash> [0..1]
   'The ash content of the coal product.'

   <sulfur> CoalAttributePercentage </sulfur> [0..1]
   'The sulfur/sulphur content of the coal product.'

   <SO2> CoalAttributePercentage </SO2> [0..1]
   'The sulfur/sulphur dioxide content of the coal product.'

   <volatile> CoalAttributePercentage </volatile> [0..1]
   'The volatile content of the coal product.'

   <BTUperLB> CoalAttributeDecimal </BTUperLB> [0..1]
   'The number of British Thermal Units per Pound of the coal product.'

   <topSize> CoalAttributeDecimal </topSize> [0..1]
   'The smallest sieve opening that will result in less than 5% of a sample of the
   coal product remaining.'

   <finesPassingScreen> CoalAttributeDecimal </finesPassingScreen> [0..1]
   <grindability> CoalAttributeDecimal </grindability> [0..1]
   'The Hardgrove Grindability Index value of the coal to be delivered.'

   <ashFusionTemperature> CoalAttributeDecimal </ashFusionTemperature> [0..1]
   'The temperature at which the ash form of the coal product fuses completely in
   accordance with the ASTM International D1857 Standard Test Methodology.'

   <initialDeformation> CoalAttributeDecimal </initialDeformation> [0..1]
   'The temperature at which an ash cone shows evidence of deformation.'

   <softeningHeightWidth> CoalAttributeDecimal </softeningHeightWidth> [0..1]
   'The temperature at which the height of an ash cone equals its width.
   (Softening temperature).'

   <softeningHeightHalfWidth> CoalAttributeDecimal </softeningHeightHalfWidth> [0..1]
   'The temperature at which the height of an ash cone equals half its width.
   (Hemisphere temperature).'

   <fluid> CoalAttributeDecimal </fluid> [0..1]
   'The temperature at which the ash cone flattens.'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalStandardQuality">
   <xsd:sequence>
      <xsd:group ref=" CommodityCoalComposition.model "/>
      <xsd:group ref=" CommodityCoalProperties.model "/>
      <xsd:group ref=" CommodityCoalReducingAtmosphere.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalStandardQualitySchedule

[Table of contents]

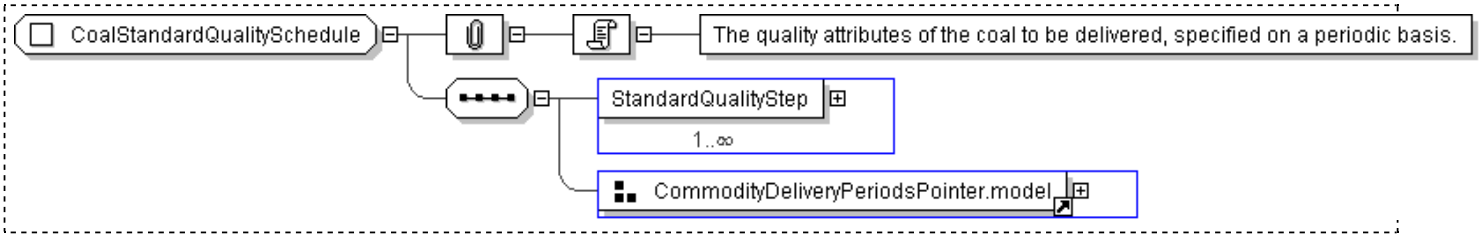
Super-types:	None
Sub-types:	None

Name	CoalStandardQualitySchedule
Used by (from the same schema document)	Complex Type CoalProductSpecifications
Abstract	no
Documentation	The quality attributes of the coal to be delivered, specified on a periodic basis.

XML Instance Representation

```
<...>
   <StandardQualityStep> CoalStandardQuality </StandardQualityStep> [1..*]
   Start Choice [1]
      <deliveryPeriodsReference> CalculationPeriodsReference </deliveryPeriodsReference> [1]
      'A pointer style reference to the Delivery Periods defined elsewhere.'

      <deliveryPeriodsScheduleReference> CalculationPeriodsScheduleReference
      </deliveryPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined elsewhere.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalStandardQualitySchedule">
   <xsd:sequence>
      <xsd:element name="StandardQualityStep" type=" CoalStandardQuality " maxOccurs="unbounded"/>
      <xsd:group ref=" CommodityDeliveryPeriodsPointer.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CoalTransportationEquipment

[Table of contents]

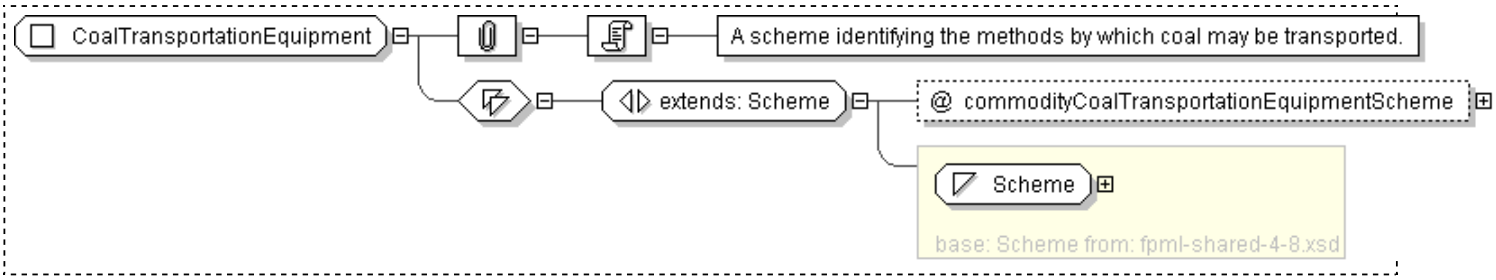
Super-types:	Scheme < CoalTransportationEquipment (by extension)
Sub-types:	None

Name	CoalTransportationEquipment
Used by (from the same schema document)	Model Group CommodityUSCoalDelivery.model
Abstract	no
Documentation	A scheme identifying the methods by which coal may be transported.

XML Instance Representation

```
<...
 commodityCoalTransportationEquipmentScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CoalTransportationEquipment">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="commodityCoalTransportationEquipmentScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/commodity-coal-transportation-equipment"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityAmericanExercise

[Table of contents]

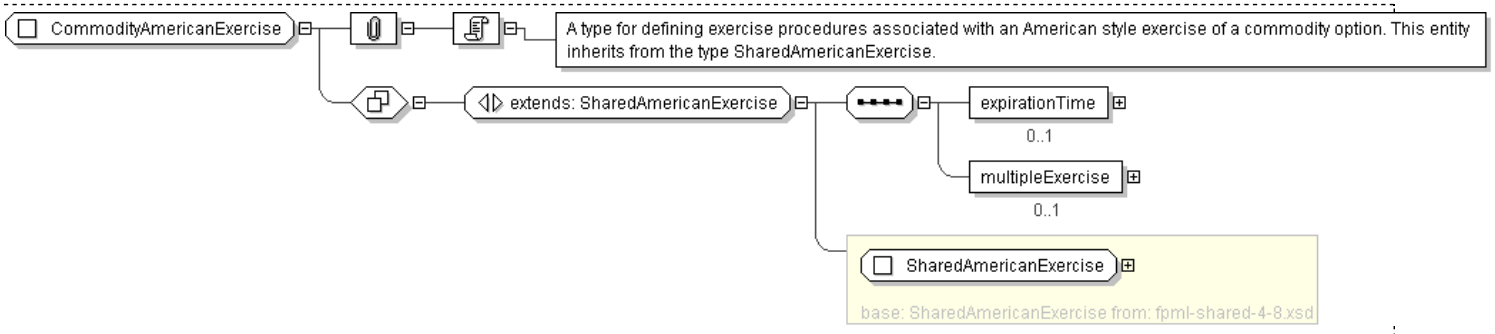
Super-types:	SharedAmericanExercise < CommodityAmericanExercise (by extension)
Sub-types:	None

Name	CommodityAmericanExercise
Used by (from the same schema document)	Complex Type CommodityExercise
Abstract	no
Documentation	A type for defining exercise procedures associated with an American style exercise of a commodity option. This entity inherits from the type SharedAmericanExercise.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
    'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
    'The last day within an exercise period for an American style option. For a European style option it is the only
    day within the exercise period.'

   Start Choice [0..1]
'Choice between latest exercise time expressed as literal time, or using a determination method.'


      <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [1]
       'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration
       date) within the exercise period that notice can be given by the buyer to the seller or seller\'s agent. Notice
       of exercise given after this time will be deemed to have been given on the next exercise business day.'

      <latestExerciseTimeDetermination> DeterminationMethod </latestExerciseTimeDetermination> [1]
       'Latest exercise time determination method.'

   End Choice
   <expirationTime> BusinessCenterTime </expirationTime> [0..1]
    'The specific time of day on which the option expires.'

   <multipleExercise> CommodityMultipleExercise </multipleExercise> [0..1]
    'The presence of this element indicates that the option may be partially exercised. It is not applicable to
    European or Asian options.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityAmericanExercise">
  <xsd:complexContent>
    <xsd:extension base=" SharedAmericanExercise ">
      <xsd:sequence>
        <xsd:element name="expirationTime" type=" BusinessCenterTime " minOccurs="0"/>
        <xsd:element name="multipleExercise" type=" CommodityMultipleExercise " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityCalculationPeriodsSchedule

[Table of contents]

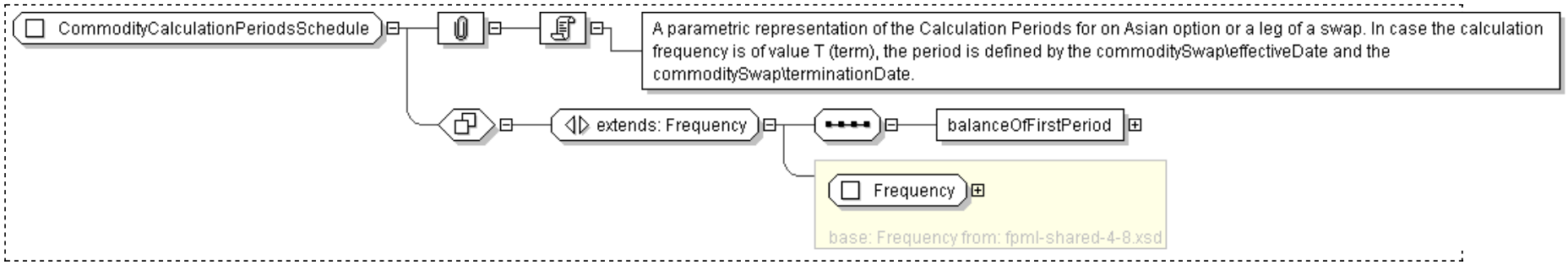
Super-types:	Frequency < CommodityCalculationPeriodsSchedule (by extension)
Sub-types:	None

Name	CommodityCalculationPeriodsSchedule
Used by (from the same schema document)	Complex Type CommodityDeliveryPeriods , Model Group CommodityAsian.model , Model Group CommodityCalculationPeriods.model
Abstract	no
Documentation	A parametric representation of the Calculation Periods for on Asian option or a leg of a swap. In case the calculation frequency is of value T (term), the period is defined by the commoditySwap\effectiveDate and the commoditySwap\terminationDate.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
    'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to another
    date, e.g. -2 days. If the period value is T (Term) then periodMultiplier must contain the value 1.'

   <period> PeriodExtendedEnum </period> [1]
    'A time period, e.g. a day, week, month, year or term of the stream. If the periodMultiplier value is 0 (zero) then
    period must contain the value D (day).'

   <balanceOfFirstPeriod> xsd:boolean </balanceOfFirstPeriod> [1]
    'If true, indicates that that the first Calculation Period should run from the Effective Date to the end of the calendar
    period in which the Effective Date falls, e.g. Jan 15 - Jan 31 if the calculation periods are one month long and
    Effective Date is Jan 15. If false, the first Calculation Period should run from the Effective Date for one whole period,
    e.g. Jan 15 to Feb 14 if the calculation periods are one month long and Effective Date is Jan 15.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityCalculationPeriodsSchedule">
  <xsd:complexContent>
    <xsd:extension base=" Frequency ">
```

```
        <xsd:sequence>
           <xsd:element name="balanceOfFirstPeriod" type=" xsd:boolean "/>
        </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityDeliveryPeriods

[Table of contents]

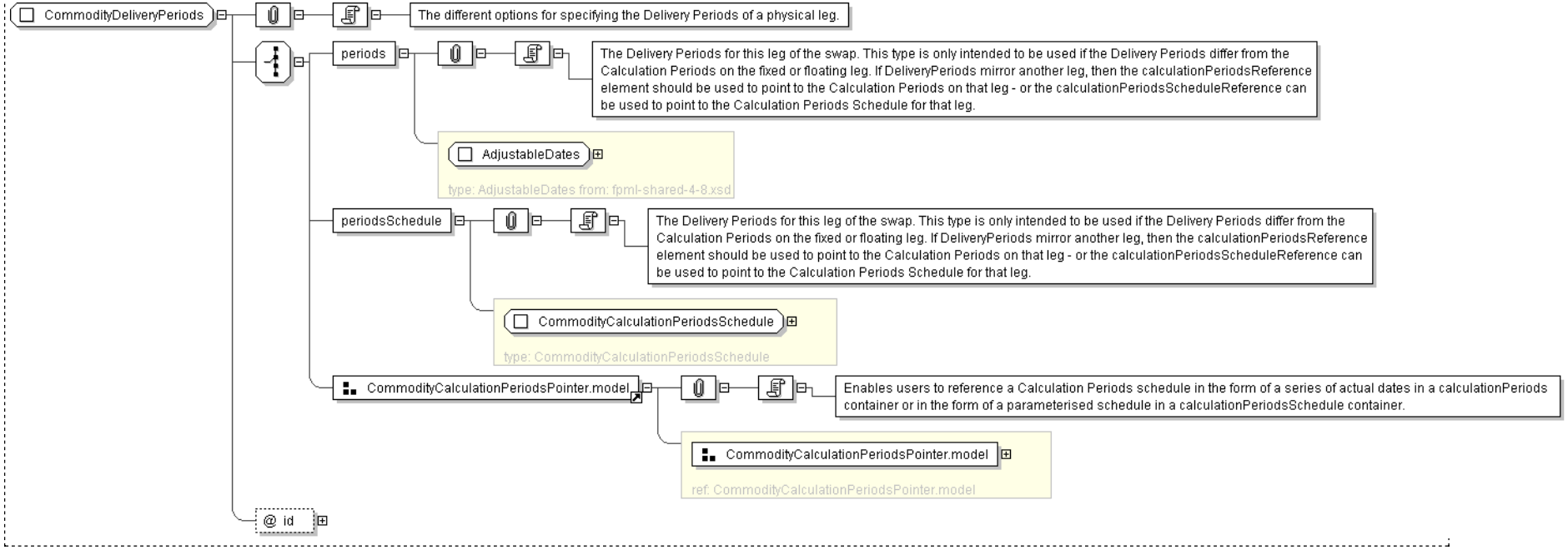
Super-types:	None
Sub-types:	• ElectricityDeliveryPeriods (by extension) • GasDeliveryPeriods (by extension)

Name	CommodityDeliveryPeriods
Used by (from the same schema document)	Complex Type CoalPhysicalLeg , Complex Type ElectricityPhysicalLeg , Complex Type OilPhysicalLeg
Abstract	no
Documentation	The different options for specifying the Delivery Periods of a physical leg.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <periods> AdjustableDates </periods> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if the Delivery Periods differ from the Calculation Periods on the fixed
      or floating leg. If DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be used to point to the Calculation Periods on that
      leg - or the calculationPeriodsScheduleReference can be used to point to the Calculation Periods Schedule for that leg.'

      <periodsSchedule> CommodityCalculationPeriodsSchedule </periodsSchedule> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if the Delivery Periods differ from the Calculation Periods on the fixed
      or floating leg. If DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be used to point to the Calculation Periods on that
      leg - or the calculationPeriodsScheduleReference can be used to point to the Calculation Periods Schedule for that leg.'

      Start Choice [1]
         <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
         'A pointer style reference to the Calculation Periods defined on another leg.'

         <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
         'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

         <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
         'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

      End Choice
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityDeliveryPeriods">
  <xsd:choice>
    <xsd:element name="periods" type=" AdjustableDates "/>
    <xsd:element name="periodsSchedule" type=" CommodityCalculationPeriodsSchedule "/>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityDeliveryPoint

[Table of contents]

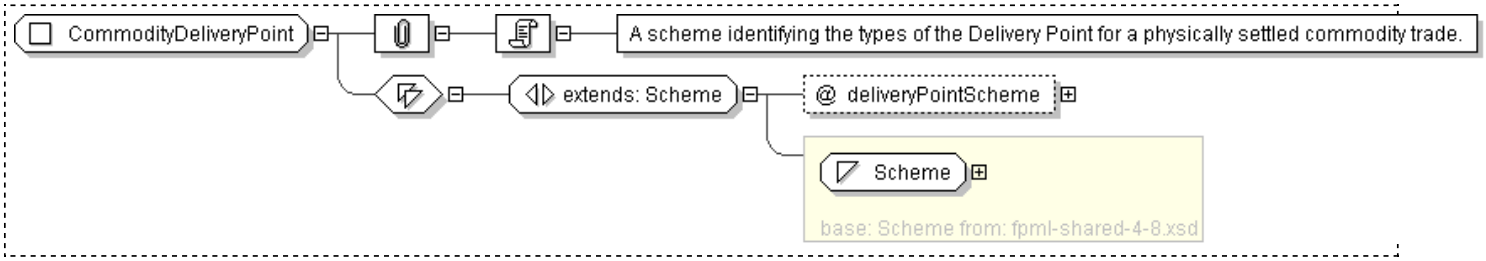
Super-types:	Scheme < CommodityDeliveryPoint (by extension)
Sub-types:	None

Name	CommodityDeliveryPoint
Used by (from the same schema document)	Complex Type ElectricityDelivery , Complex Type ElectricityDeliverySystemFirm , Complex Type ElectricityDeliveryUnitFirm , Complex Type OilPipelineDelivery , Complex Type OilPipelineDelivery , Complex Type OilTransferDelivery , Model Group CommodityDeliveryPoints.model , Model Group CommodityDeliveryPoints.model
Abstract	no
Documentation	A scheme identifying the types of the Delivery Point for a physically settled commodity trade.

XML Instance Representation

```
<...
 deliveryPointScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityDeliveryPoint">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="deliveryPointScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityDeliveryRisk

[Table of contents]

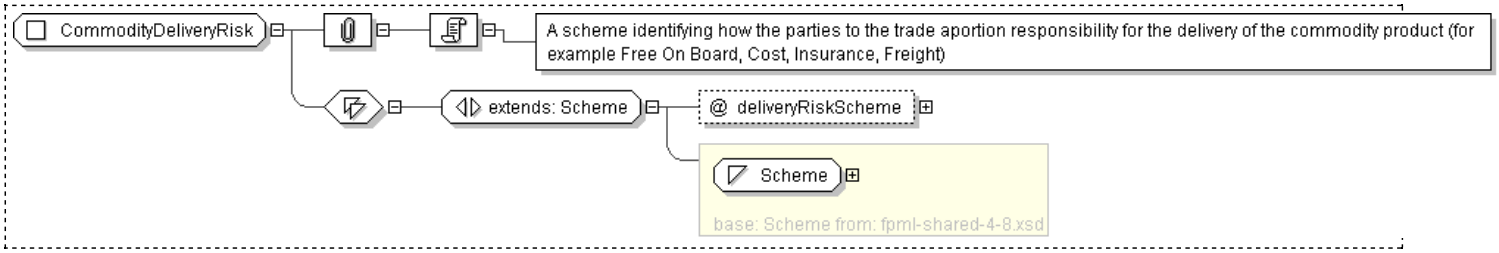
Super-types:	Scheme < CommodityDeliveryRisk (by extension)
Sub-types:	None

Name	CommodityDeliveryRisk
Used by (from the same schema document)	Complex Type OilPipelineDelivery , Model Group CommodityUSCoalDelivery.model
Abstract	no
Documentation	A scheme identifying how the parties to the trade aportion responsibility for the delivery of the commodity product (for example Free On Board, Cost, Insurance, Freight)

XML Instance Representation

```
<...
 deliveryRiskScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityDeliveryRisk">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="deliveryRiskScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/external/incoterms"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityEuropeanExercise

[Table of contents]

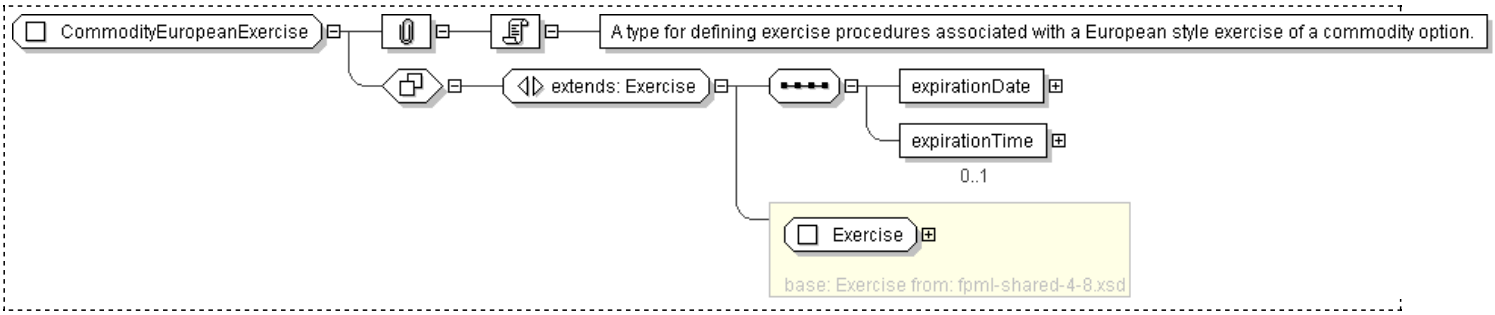
Super-types:	Exercise < CommodityEuropeanExercise (by extension)
Sub-types:	None

Name	CommodityEuropeanExercise
Used by (from the same schema document)	Complex Type CommodityExercise
Abstract	no
Documentation	A type for defining exercise procedures associated with a European style exercise of a commodity option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the
   only day within the exercise period. For an averaging option this is equivalent to the Termination Date.'

   <expirationTime> BusinessCenterTime </expirationTime> [0..1]
   'The specific time of day on which the option expires.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityEuropeanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="expirationTime" type=" BusinessCenterTime " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityExercise

[Table of contents]

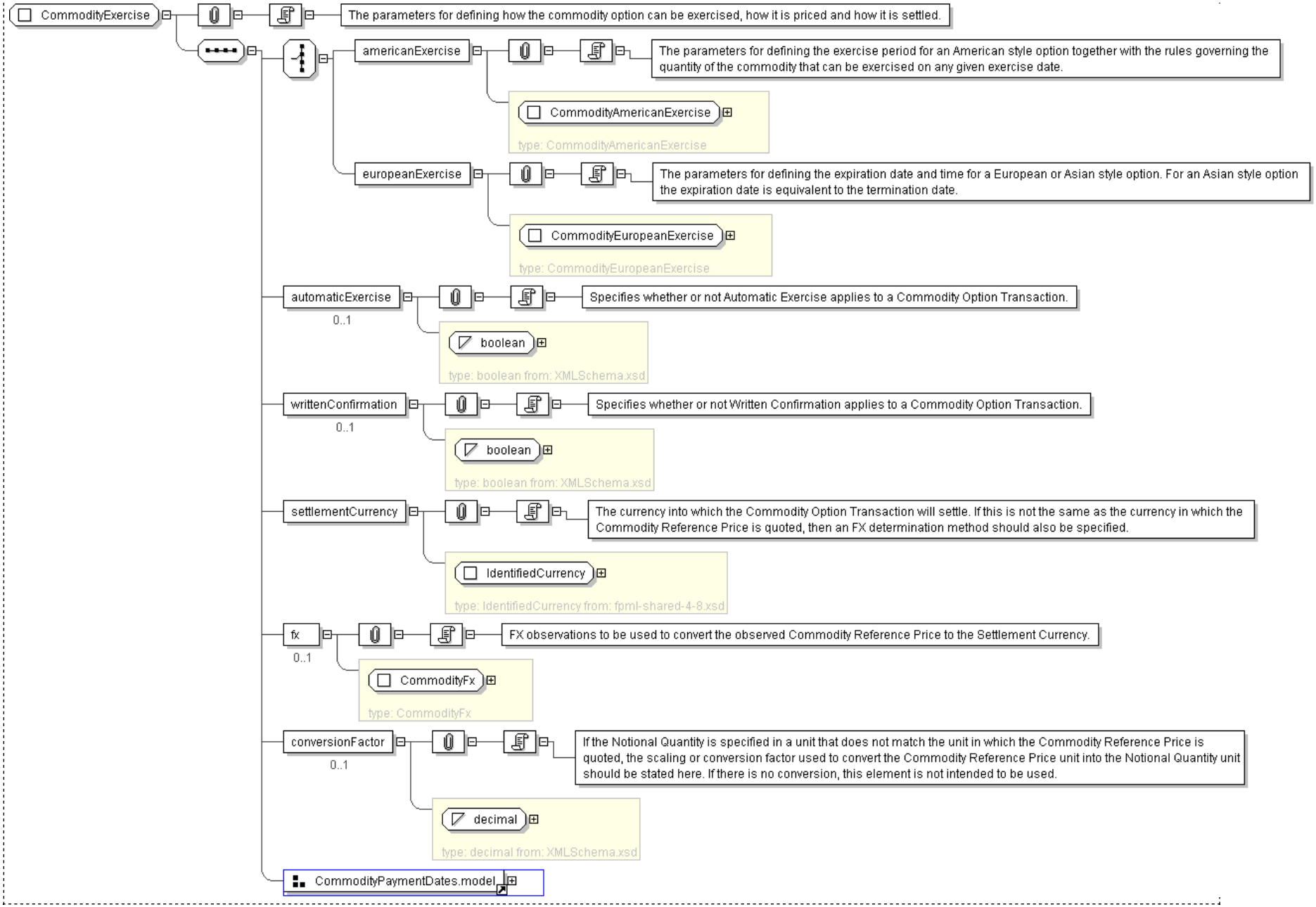
Super-types:	None
Sub-types:	None

Name	CommodityExercise
Used by (from the same schema document)	Model Group CommodityFinancialOption.model
Abstract	no
Documentation	The parameters for defining how the commodity option can be exercised, how it is priced and how it is settled.

XML Instance Representation

```
<...>
   Start Choice [1]
      <americanExercise> CommodityAmericanExercise </americanExercise> [1]
      'The parameters for defining the exercise period for an American style option together with the rules governing the quantity of the commodity
      that can be exercised on any given exercise date.'

      <europeanExercise> CommodityEuropeanExercise </europeanExercise> [1]
      'The parameters for defining the expiration date and time for a European or Asian style option. For an Asian style option the expiration date
      is equivalent to the termination date.'

   End Choice
   <automaticExercise> xsd:boolean </automaticExercise> [0..1]
   'Specifies whether or not Automatic Exercise applies to a Commodity Option Transaction.'

   <writtenConfirmation> xsd:boolean </writtenConfirmation> [0..1]
   'Specifies whether or not Written Confirmation applies to a Commodity Option Transaction.'

   <settlementCurrency> IdentifiedCurrency </settlementCurrency> [1]
   'The currency into which the Commodity Option Transaction will settle. If this is not the same as the currency in which the Commodity Reference
   Price is quoted, then an FX determination method should also be specified.'

   <fx> CommodityFx </fx> [0..1]
   'FX observations to be used to convert the observed Commodity Reference Price to the Settlement Currency.'

   <conversionFactor> xsd:decimal </conversionFactor> [0..1]
   'If the Notional Quantity is specified in a unit that does not match the unit in which the Commodity Reference Price is quoted, the scaling or
   conversion factor used to convert the Commodity Reference Price unit into the Notional Quantity unit should be stated here. If there is no
   conversion, this element is not intended to be used.'

   Start Choice [1]
      <relativePaymentDates> CommodityRelativePaymentDates </relativePaymentDates> [1]
      'The Payment Dates of the trade relative to the Calculation Periods.'

      Start Choice [1]
         <paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]
         'Dates on which payments will be made.'

         <masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]
         'If present and true indicates that the Payment Date(s) are specified in the relevant master agreement.'

      End Choice
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityExercise">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="americanExercise" type=" CommodityAmericanExercise "/>
      <xsd:element name="europeanExercise" type=" CommodityEuropeanExercise "/>
    </xsd:choice>
    <xsd:element name="automaticExercise" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="writtenConfirmation" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="settlementCurrency" type=" IdentifiedCurrency "/>
    <xsd:element name="fx" type=" CommodityFx " minOccurs="0"/>
    <xsd:element name="conversionFactor" type=" xsd:decimal " minOccurs="0"/>
```

```
      <xsd:group ref=" CommodityPaymentDates.model "/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityExpireRelativeToEvent

[Table of contents]

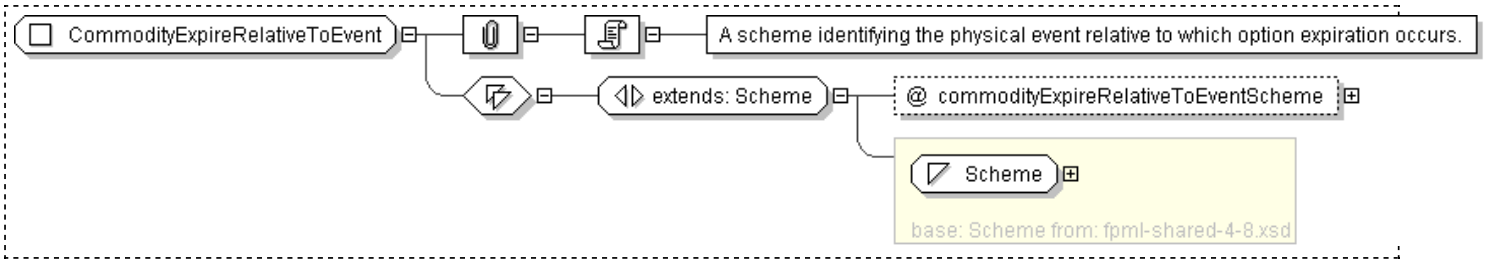
Super-types:	Scheme < CommodityExpireRelativeToEvent (by extension)
Sub-types:	None

Name	CommodityExpireRelativeToEvent
Used by (from the same schema document)	Complex Type CommodityRelativeExpirationDates
Abstract	no
Documentation	A scheme identifying the physical event relative to which option expiration occurs.

XML Instance Representation

```
<...
 commodityExpireRelativeToEventScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityExpireRelativeToEvent">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityExpireRelativeToEventScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-expire-relative-to-event"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityFixedPriceSchedule

[Table of contents]

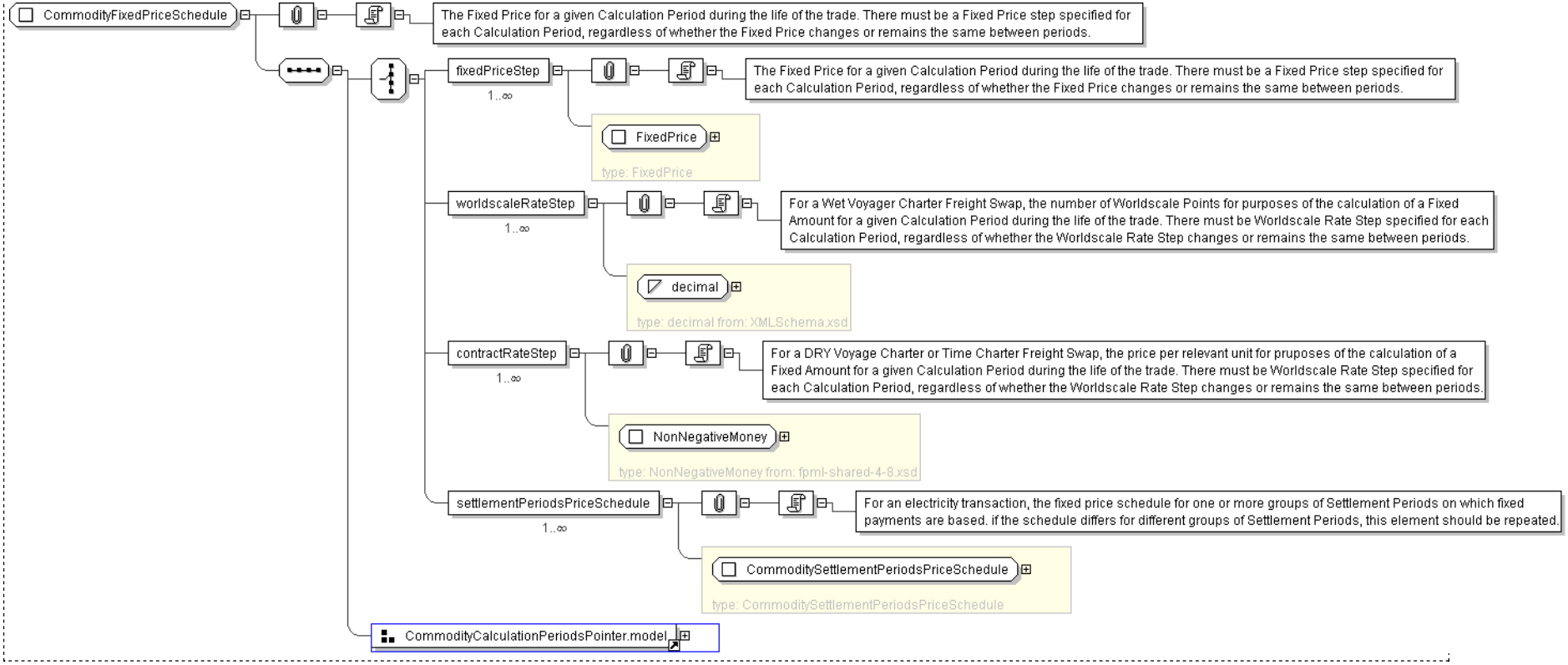
Super-types:	None
Sub-types:	None

Name	CommodityFixedPriceSchedule
Used by (from the same schema document)	Model Group CommodityFixedPrice.model
Abstract	no
Documentation	The Fixed Price for a given Calculation Period during the life of the trade. There must be a Fixed Price step specified for each Calculation Period, regardless of whether the Fixed Price changes or remains the same between periods.

XML Instance Representation

```
<...>
  Start Choice [1]
    <fixedPriceStep> FixedPrice </fixedPriceStep> [1..*]
      'The Fixed Price for a given Calculation Period during the life of the trade. There must be a Fixed Price step specified for each Calculation Period,
      regardless of whether the Fixed Price changes or remains the same between periods.'

    <worldscaleRateStep> xsd:decimal </worldscaleRateStep> [1..*]
      'For a Wet Voyager Charter Freight Swap, the number of Worldscale Points for purposes of the calculation of a Fixed Amount for a given Calculation Period during
      the life of the trade. There must be Worldscale Rate Step specified for each Calculation Period, regardless of whether the Worldscale Rate Step changes or
      remains the same between periods.'

    <contractRateStep> NonNegativeMoney </contractRateStep> [1..*]
      'For a DRY Voyage Charter or Time Charter Freight Swap, the price per relevant unit for pruposes of the calculation of a Fixed Amount for a given Calculation
      Period during the life of the trade. There must be Worldscale Rate Step specified for each Calculation Period, regardless of whether the Worldscale Rate Step
      changes or remains the same between periods.'

    <settlementPeriodsPriceSchedule> CommoditySettlementPeriodsPriceSchedule </settlementPeriodsPriceSchedule> [1..*]
      'For an electricity transaction, the fixed price schedule for one or more groups of Settlement Periods on which fixed payments are based. if the schedule
      differs for different groups of Settlement Periods, this element should be repeated.'

  End Choice
  Start Choice [1]
    <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

    <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

    <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityFixedPriceSchedule">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="fixedPriceStep" type=" FixedPrice " maxOccurs="unbounded"/>
      <xsd:element name="worldscaleRateStep" type=" xsd:decimal " maxOccurs="unbounded"/>
      <xsd:element name="contractRateStep" type=" NonNegativeMoney " maxOccurs="unbounded"/>
      <xsd:element name="settlementPeriodsPriceSchedule" type=" CommoditySettlementPeriodsPriceSchedule " maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityForward

[Table of contents]

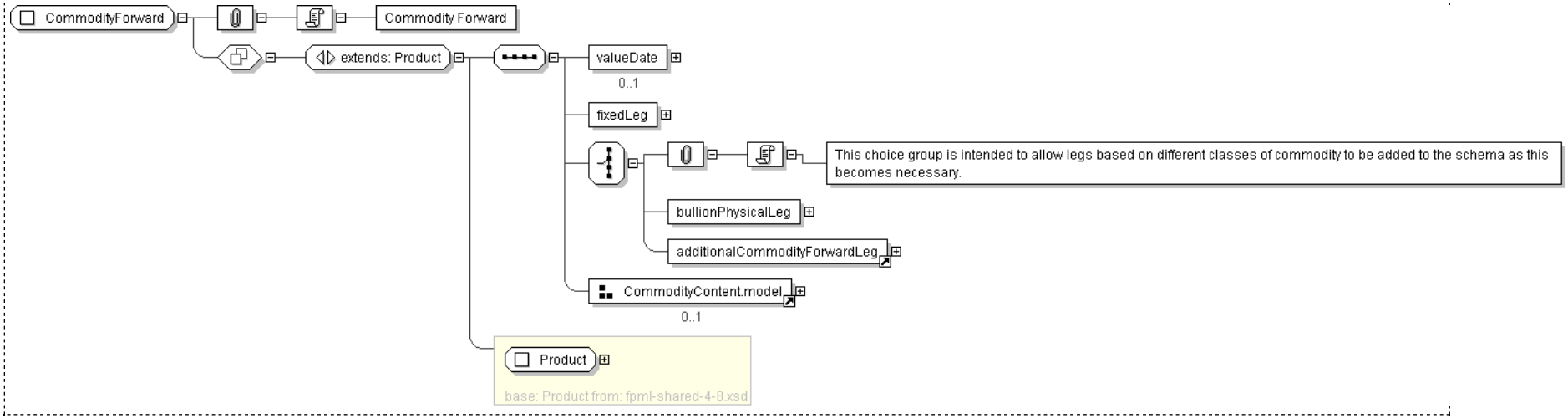
Super-types:	Product < CommodityForward (by extension)
Sub-types:	None

Name	CommodityForward
Used by (from the same schema document)	Element commodityForward
Abstract	no
Documentation	Commodity Forward

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <valueDate> AdjustableOrRelativeDate </valueDate> [0..1]
    'Specifies the value date of the Commodity Forward Transaction. This is the day on which both the cash and the physical commodity settle.'

   <fixedLeg> NonPeriodicFixedPriceLeg </fixedLeg> [1]
    'The fixed leg of a Commodity Forward Transaction'

   Start Choice [1]
 'This choice group is intended to allow legs based on different classes of commodity to be added to the schema as this becomes necessary.'


      <bullionPhysicalLeg> BullionPhysicalLeg </bullionPhysicalLeg> [1]
       'The physical leg of a Commodity Forward Transaction for which the underlyer is Bullion.'

      <additionalCommodityForwardLeg> ... </additionalCommodityForwardLeg> [1]
   End Choice
   Start Group: CommodityContent.model [0..1]
      <commonPricing> xsd:boolean </commonPricing> [0..1]
       'Common pricing may be relevant for a Transaction that references more than one Commodity Reference Price. If Common Pricing is not specified as
       applicable, it will be deemed not to apply.'

      <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
       'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA 2005 Commodity Definitions, as applicable.'

      <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption> [0..1]
       'The consequences of Bullion Settlement Disruption Events.'

      <rounding> Rounding </rounding> [0..1]
       'Rounding direction and precision for amounts.'

   End Group: CommodityContent.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityForward">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="valueDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
        <xsd:element name="fixedLeg" type=" NonPeriodicFixedPriceLeg "/>
        <xsd:choice>
          <xsd:element name="bullionPhysicalLeg" type=" BullionPhysicalLeg "/>
          <xsd:element ref=" additionalCommodityForwardLeg "/>
        </xsd:choice>
        <xsd:group ref=" CommodityContent.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityFrequencyType

[Table of contents]

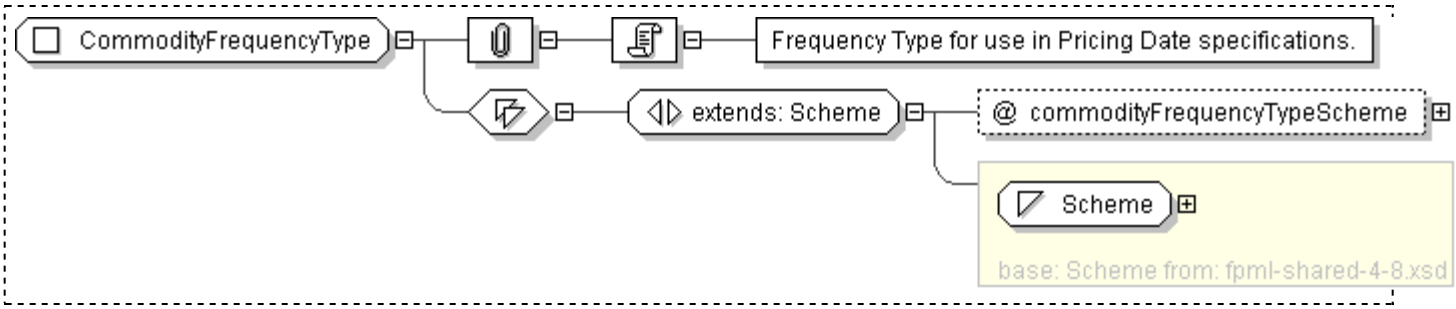
Super-types:	Scheme < CommodityFrequencyType (by extension)
Sub-types:	None

Name	CommodityFrequencyType
Used by (from the same schema document)	Model Group PricingDays.model
Abstract	no
Documentation	Frequency Type for use in Pricing Date specifications.

XML Instance Representation

```
<...
 commodityFrequencyTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityFrequencyType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityFrequencyTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-frequency-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityFx

[Table of contents]

Super-types:	None
Sub-types:	None

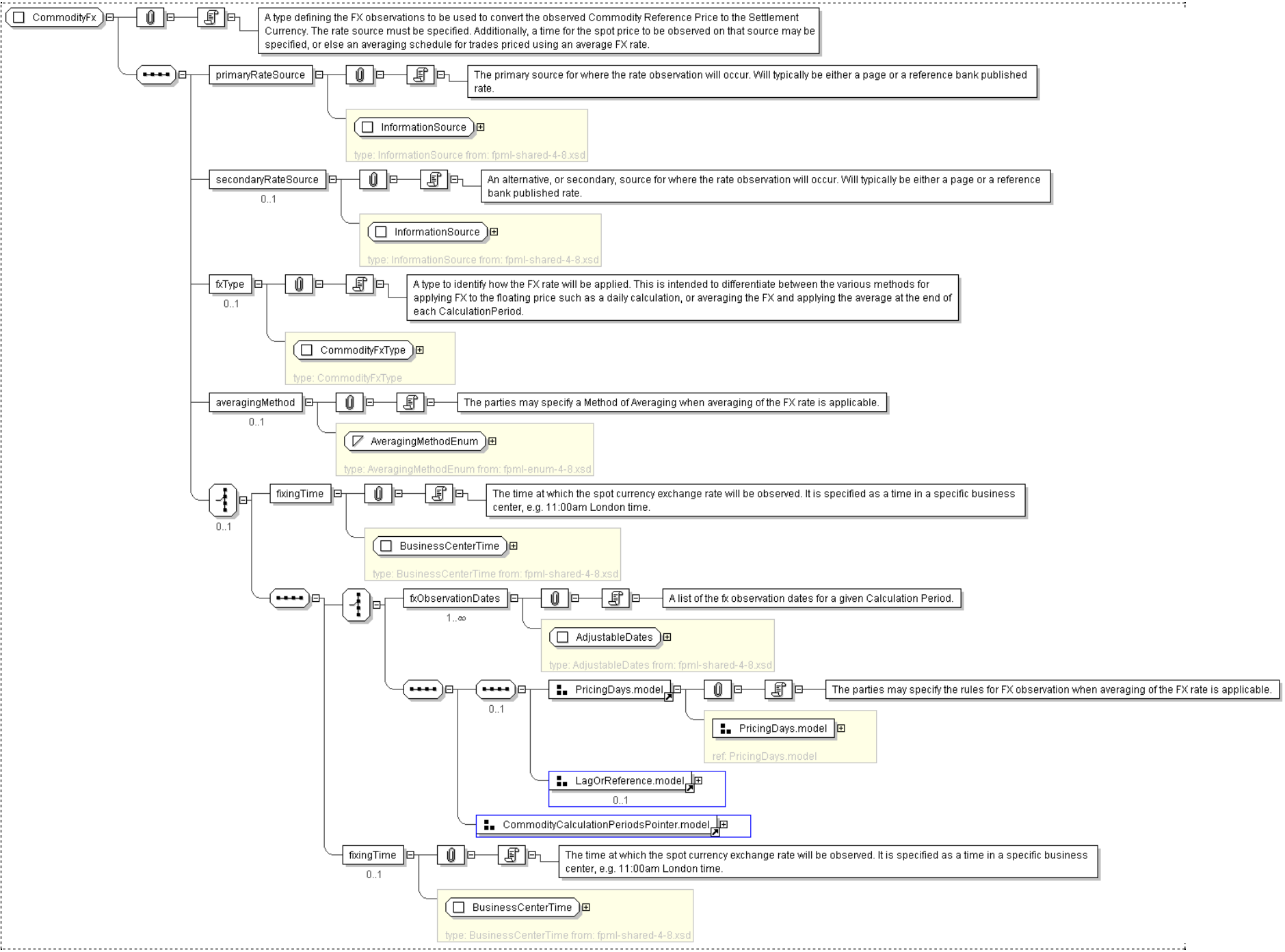
Name	CommodityFx
Used by (from the same schema document)	Complex Type CommodityExercise , Complex Type FloatingLegCalculation
Abstract	no
Documentation	A type defining the FX observations to be used to convert the observed Commodity Reference Price to the Settlement Currency. The rate source must be specified. Additionally, a time for the spot price to be observed on that source may be specified, or else an averaging schedule for trades priced using an average FX rate.

XML Instance Representation

```
<...>
   <primaryRateSource> InformationSource </primaryRateSource> [1]
   'The primary source for where the rate observation will occur. Will typically be either a page or a reference bank published rate.'

   <secondaryRateSource> InformationSource </secondaryRateSource> [0..1]
   'An alternative, or secondary, source for where the rate observation will occur. Will typically be either a page or a reference bank published rate.'

   <fxType> CommodityFxType </fxType> [0..1]
   'A type to identify how the FX rate will be applied. This is intended to differentiate between the various methods for applying FX to the floating price such as a daily
   calculation, or averaging the FX and applying the average at the end of each CalculationPeriod.'

   <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
   'The parties may specify a Method of Averaging when averaging of the FX rate is applicable.'

   Start Choice [0..1]
      <fixingTime> BusinessCenterTime </fixingTime> [1]
      'The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g. 11:00am London time.'

      Start Choice [1]
         <fxObservationDates> AdjustableDates </fxObservationDates> [1..*]
         'A list of the fx observation dates for a given Calculation Period.'

         Start Sequence [0..1]
            <dayType> CommodityDayTypeEnum </dayType> [1]
            'The type of day on which pricing occurs.'

            Start Choice [1]
               <dayDistribution> CommodityFrequencyType </dayDistribution> [1]
               'The method by which the pricing days are distributed across the pricing period.'

               <dayCount> xsd:positiveInteger </dayCount> [0..1]
               'The number of days over which pricing should take place.'

               <dayOfWeek> DayOfWeekEnum </dayOfWeek> [1..7]
               'The day(s) of the week on which pricing will take place during the pricing period.'

               <dayNumber> xsd:integer </dayNumber> [0..1]
               'The occurrence of the dayOfWeek within the pricing period on which pricing will take place, e.g. the 3rd Friday within each Calculation Period. If omitted,
               every dayOfWeek will be a pricing day.'

            End Choice
            Start Group: LagOrReference.model [0..1]
               Start Choice [1]
                  <lag> Lag </lag> [1]
                  'The pricing period per calculation period if the pricing days do not wholly fall within the respective calculation period.'

                  <lagReference> LagReference </lagReference> [1]
                  'Allows a lag to reference one already defined elsewhere in the trade.'

               End Choice
            End Group: LagOrReference.model
         End Sequence
         Start Choice [1]
            <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
            'A pointer style reference to the Calculation Periods defined on another leg.'

            <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
            'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

            <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
            'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

         End Choice
      End Choice
      <fixingTime> BusinessCenterTime </fixingTime> [0..1]
```

```
    'The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g. 11:00am London time.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityFx">
  <xsd:sequence>
    <xsd:element name="primaryRateSource" type=" InformationSource "/>
    <xsd:element name="secondaryRateSource" type=" InformationSource " minOccurs="0"/>
    <xsd:element name="fxType" type=" CommodityFxType " minOccurs="0"/>
    <xsd:element name="averagingMethod" type=" AveragingMethodEnum " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="fixingTime" type=" BusinessCenterTime "/>
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="fxObservationDates" type=" AdjustableDates " maxOccurs="unbounded"/>
          <xsd:sequence>
            <xsd:sequence minOccurs="0">
              <xsd:group ref=" PricingDays.model "/>
              <xsd:group ref=" LagOrReference.model " minOccurs="0"/>
            </xsd:sequence>
            <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
          </xsd:sequence>
        </xsd:choice>
        <xsd:element name="fixingTime" type=" BusinessCenterTime " minOccurs="0"/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityFxType

[Table of contents]

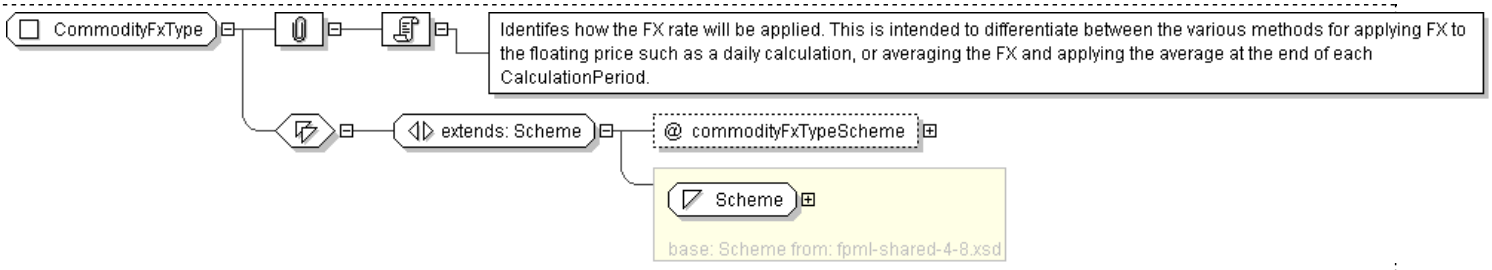
Super-types:	Scheme < CommodityFxType (by extension)
Sub-types:	None

Name	CommodityFxType
Used by (from the same schema document)	Complex Type CommodityFx
Abstract	no
Documentation	Identifes how the FX rate will be applied. This is intended to differentiate between the various methods for applying FX to the floating price such as a daily calculation, or averaging the FX and applying the average at the end of each CalculationPeriod.

XML Instance Representation

```
<...
 commodityFxTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityFxType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="commodityFxTypeScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/commodity-fx-type"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityHub

[Table of contents]

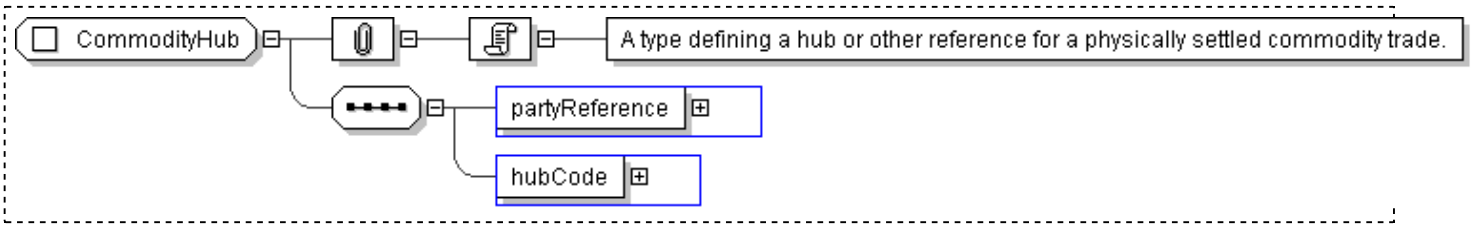
Super-types:	None
Sub-types:	None

Name	CommodityHub
Used by (from the same schema document)	Complex Type GasDelivery , Complex Type GasDelivery
Abstract	no
Documentation	A type defining a hub or other reference for a physically settled commodity trade.

XML Instance Representation

```
<...>
   <partyReference> PartyOrAccountReference </partyReference> [1]
   <hubCode> CommodityHubCode </hubCode> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityHub">
   <xsd:sequence>
      <xsd:element name="partyReference" type=" PartyOrAccountReference "/>
      <xsd:element name="hubCode" type=" CommodityHubCode "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityHubCode

[Table of contents]

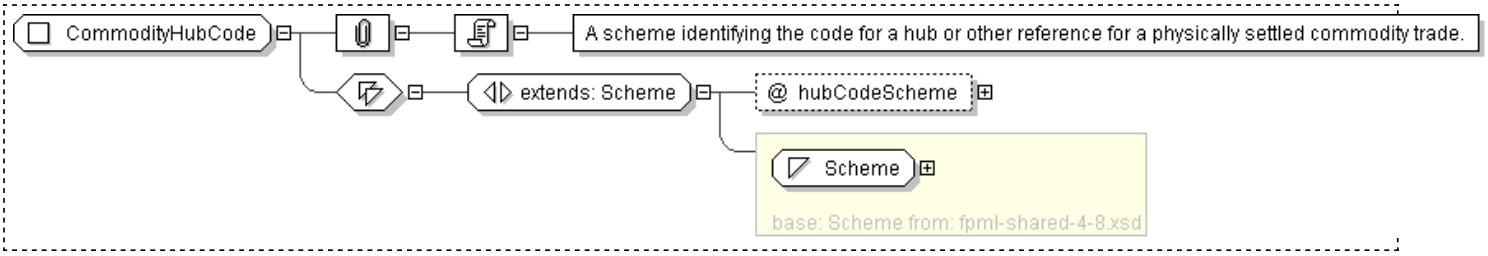
Super-types:	Scheme < CommodityHubCode (by extension)
Sub-types:	None

Name	CommodityHubCode
Used by (from the same schema document)	Complex Type CommodityHub
Abstract	no
Documentation	A scheme identifying the code for a hub or other reference for a physically settled commodity trade.

XML Instance Representation

```
<...
 hubCodeScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityHubCode">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="hubCodeScheme" type=" xsd:anyURI " use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityMarketDisruption

[Table of contents]

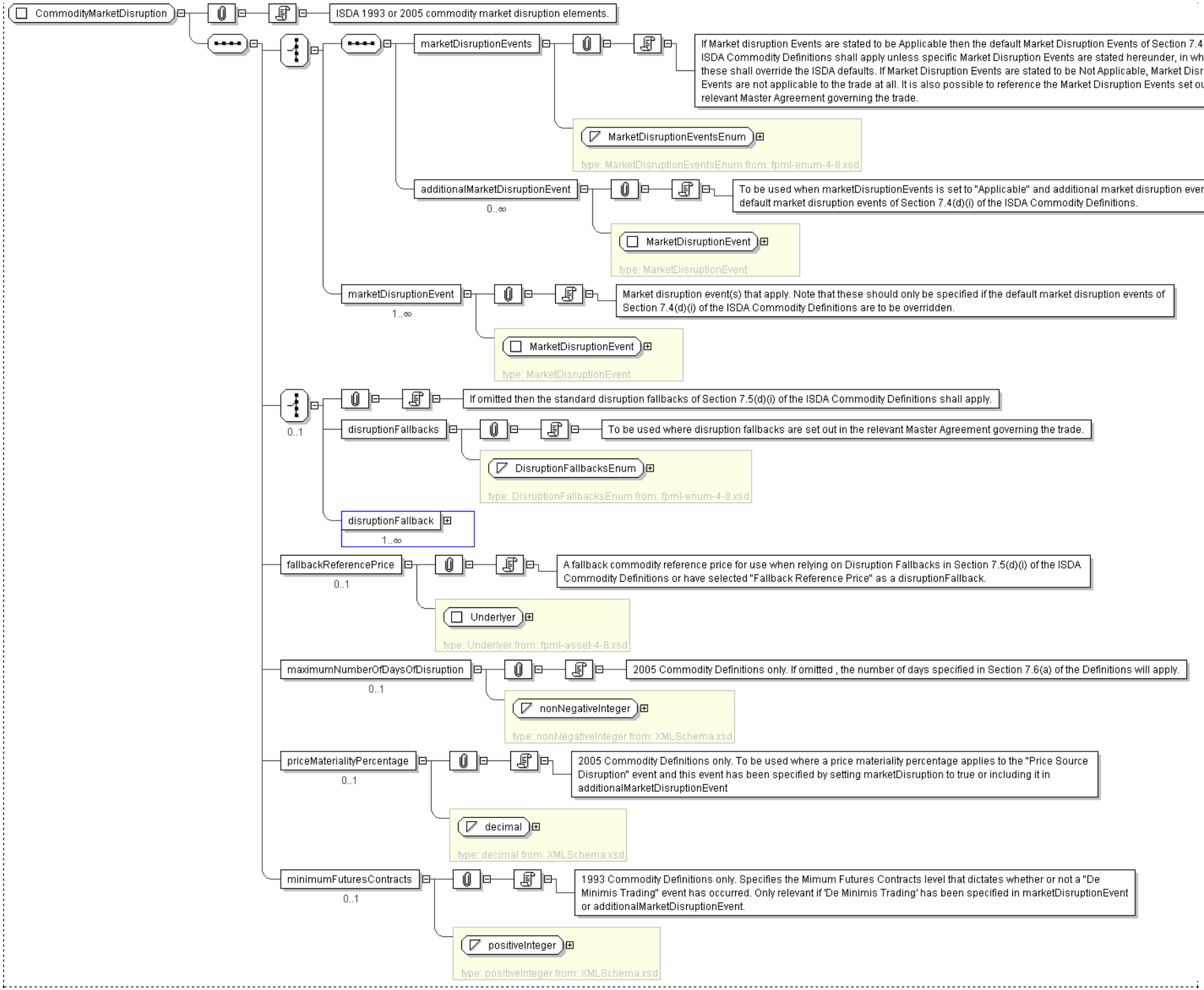
Super-types:	None
Sub-types:	None

Name	CommodityMarketDisruption
Used by (from the same schema document)	Model Group CommodityContent.model
Abstract	no
Documentation	ISDA 1993 or 2005 commodity market disruption elements.

XML Instance Representation

```
<...>
  Start Choice [1]
    <marketDisruptionEvents> MarketDisruptionEventsEnum </marketDisruptionEvents> [1]
    'If Market disruption Events are stated to be Applicable then the default Market Disruption Events of Section 7.4(d)(i) of the ISDA Commodity Definitions shall apply
    unless specific Market Disruption Events are stated hereunder, in which case these shall override the ISDA defaults. If Market Disruption Events are stated to be Not
    Applicable, Market Disruption Events are not applicable to the trade at all. It is also possible to reference the Market Disruption Events set out in the relevant
    Master Agreement governing the trade.'

    <additionalMarketDisruptionEvent> MarketDisruptionEvent </additionalMarketDisruptionEvent> [0..*]
    'To be used when marketDisruptionEvents is set to \"Applicable\" and additional market disruption events(s) apply to the default market disruption events of Section
    7.4(d)(i) of the ISDA Commodity Definitions.'

    <marketDisruptionEvent> MarketDisruptionEvent </marketDisruptionEvent> [1..*]
    'Market disruption event(s) that apply. Note that these should only be specified if the default market disruption events of Section 7.4(d)(i) of the ISDA Commodity
    Definitions are to be overridden.'

  End Choice
  Start Choice [0..1]
'If omitted then the standard disruption fallbacks of Section 7.5(d)(i) of the ISDA Commodity Definitions shall apply.'


    <disruptionFallbacks> DisruptionFallbacksEnum </disruptionFallbacks> [1]
    'To be used where disruption fallbacks are set out in the relevant Master Agreement governing the trade.'

    <disruptionFallback> SequencedDisruptionFallback </disruptionFallback> [1..*]
  End Choice
  <fallbackReferencePrice> Underlyer </fallbackReferencePrice> [0..1]
  'A fallback commodity reference price for use when relying on Disruption Fallbacks in Section 7.5(d)(i) of the ISDA Commodity Definitions or have selected \"Fallback
  Reference Price\" as a disruptionFallback.'

  <maximumNumberOfDaysOfDisruption> xsd:nonNegativeInteger </maximumNumberOfDaysOfDisruption> [0..1]
  '2005 Commodity Definitions only. If omitted , the number of days specified in Section 7.6(a) of the Definitions will apply.'

  <priceMaterialityPercentage> xsd:decimal </priceMaterialityPercentage> [0..1]
  '2005 Commodity Definitions only. To be used where a price materiality percentage applies to the \"Price Source Disruption\" event and this event has been specified by
  setting marketDisruption to true or including it in additionalMarketDisruptionEvent'

  <minimumFuturesContracts> xsd:positiveInteger </minimumFuturesContracts> [0..1]
  '1993 Commodity Definitions only. Specifies the Mimum Futures Contracts level that dictates whether or not a \"De Minimis Trading\" event has occurred. Only relevant if
  \'De Minimis Trading\' has been specified in marketDisruptionEvent or additionalMarketDisruptionEvent.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityMarketDisruption">
  <xsd:sequence>
    <xsd:choice>
      <xsd:sequence>
```

```
          <xsd:element name="marketDisruptionEvents" type=" MarketDisruptionEventsEnum "/>
          <xsd:element name="additionalMarketDisruptionEvent" type=" MarketDisruptionEvent " minOccurs="0" maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:element name="marketDisruptionEvent" type=" MarketDisruptionEvent " maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:choice minOccurs="0">
        <xsd:element name="disruptionFallbacks" type=" DisruptionFallbacksEnum "/>
        <xsd:element name="disruptionFallback" type=" SequencedDisruptionFallback " maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:element name="fallbackReferencePrice" type=" Underlyer " minOccurs="0"/>
      <xsd:element name="maximumNumberOfDaysOfDisruption" type=" xsd:nonNegativeInteger " minOccurs="0"/>
      <xsd:element name="priceMaterialityPercentage" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="minimumFuturesContracts" type=" xsd:positiveInteger " minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityMultipleExercise

[Table of contents]

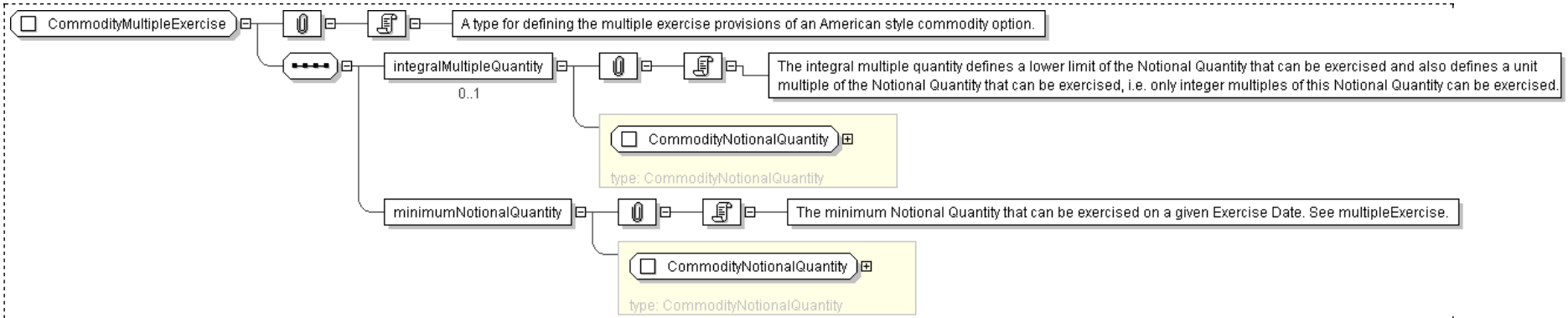
Super-types:	None
Sub-types:	None

Name	CommodityMultipleExercise
Used by (from the same schema document)	Complex Type CommodityAmericanExercise
Abstract	no
Documentation	A type for defining the multiple exercise provisions of an American style commodity option.

XML Instance Representation

```
<...>
   <integralMultipleQuantity> CommodityNotionalQuantity </integralMultipleQuantity> [0..1]
   'The integral multiple quantity defines a lower limit of the Notional Quantity that can be exercised and also defines a unit multiple of the
   Notional Quantity that can be exercised, i.e. only integer multiples of this Notional Quantity can be exercised.'

   <minimumNotionalQuantity> CommodityNotionalQuantity </minimumNotionalQuantity> [1]
   'The minimum Notional Quantity that can be exercised on a given Exercise Date. See multipleExercise.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityMultipleExercise">
  <xsd:sequence>
    <xsd:element name="integralMultipleQuantity" type=" CommodityNotionalQuantity " minOccurs="0"/>
    <xsd:element name="minimumNotionalQuantity" type=" CommodityNotionalQuantity "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityNotionalQuantity

[Table of contents]

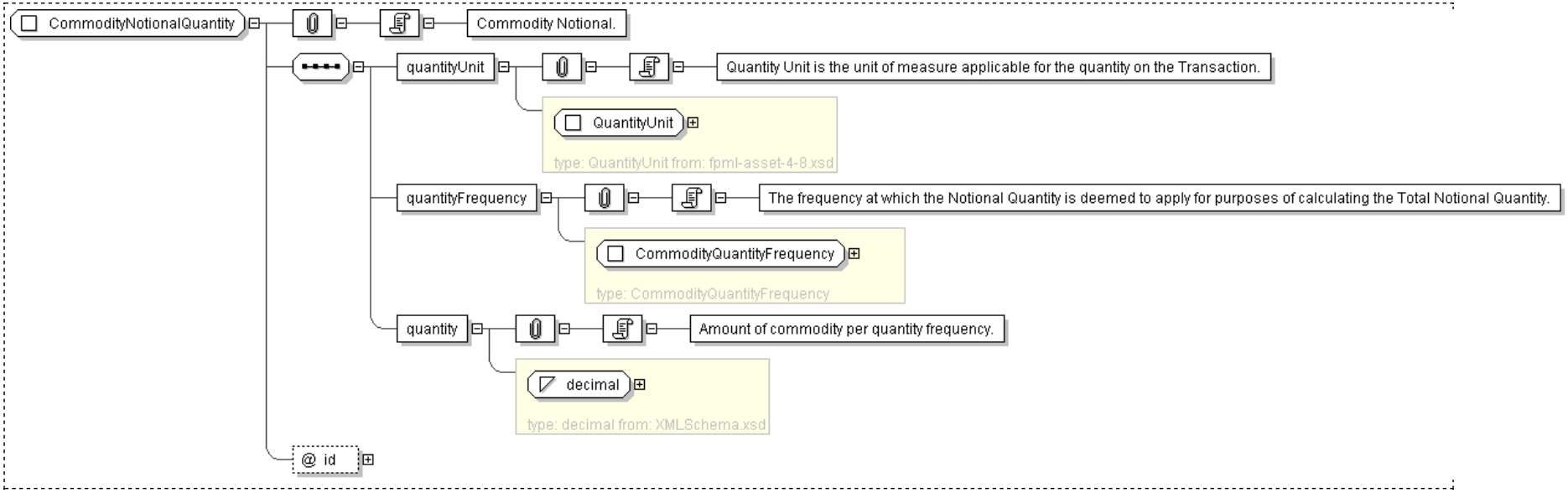
Super-types:	None
Sub-types:	• CommoditySettlementPeriodsNotionalQuantity (by extension) • ElectricityPhysicalDeliveryQuantity (by extension)

Name	CommodityNotionalQuantity
Used by (from the same schema document)	Complex Type CommodityMultipleExercise , Complex Type CommodityMultipleExercise , Complex Type CommodityNotionalQuantitySchedule , Complex Type CommodityPhysicalQuantitySchedule , Complex Type CommoditySettlementPeriodsNotionalQuantitySchedule , Complex Type GasPhysicalQuantity , Complex Type GasPhysicalQuantity , Model Group CommodityFixedPhysicalQuantity.model , Model Group CommodityNotionalQuantity.model
Abstract	no
Documentation	Commodity Notional.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityUnit> QuantityUnit </quantityUnit> [1]
   'Quantity Unit is the unit of measure applicable for the quantity on the Transaction.'

   <quantityFrequency> CommodityQuantityFrequency </quantityFrequency> [1]
   'The frequency at which the Notional Quantity is deemed to apply for purposes of calculating the Total Notional Quantity.'

   <quantity> xsd:decimal </quantity> [1]
   'Amount of commodity per quantity frequency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityNotionalQuantity">
   <xsd:sequence>
      <xsd:element name="quantityUnit" type=" QuantityUnit "/>
      <xsd:element name="quantityFrequency" type=" CommodityQuantityFrequency "/>
      <xsd:element name="quantity" type=" xsd:decimal "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityNotionalQuantitySchedule

[Table of contents]

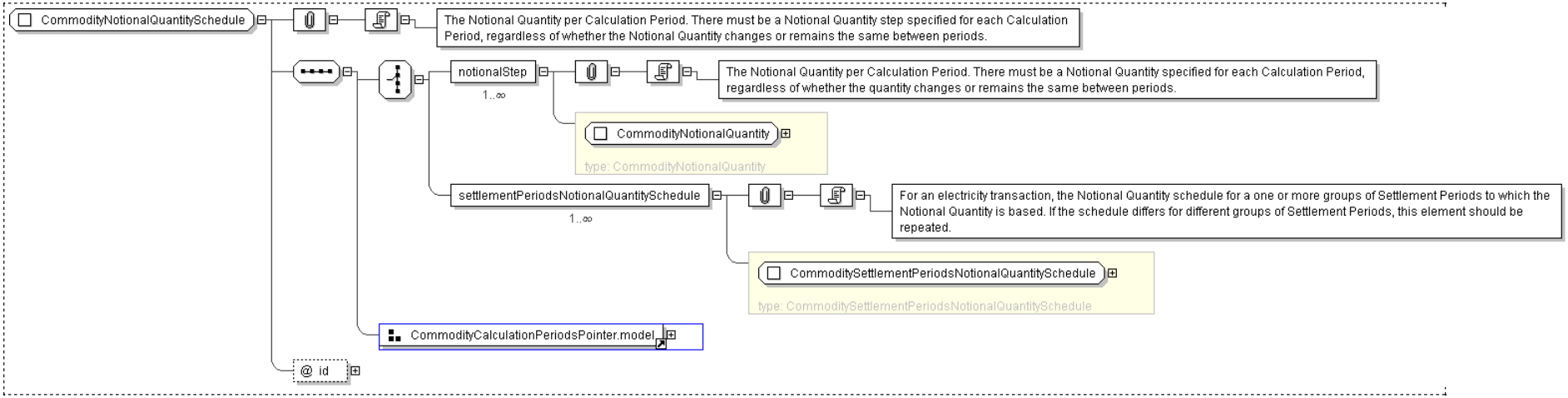
Super-types:	None
Sub-types:	None

Name	CommodityNotionalQuantitySchedule
Used by (from the same schema document)	Model Group CommodityNotionalQuantity.model
Abstract	no
Documentation	The Notional Quantity per Calculation Period. There must be a Notional Quantity step specified for each Calculation Period, regardless of whether the Notional Quantity changes or remains the same between periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <notionalStep> CommodityNotionalQuantity </notionalStep> [1..*]
      'The Notional Quantity per Calculation Period. There must be a Notional Quantity specified for each Calculation Period, regardless of whether the quantity changes or
      remains the same between periods.'

      <settlementPeriodsNotionalQuantitySchedule> CommoditySettlementPeriodsNotionalQuantitySchedule </settlementPeriodsNotionalQuantitySchedule> [1..*]
      'For an electricity transaction, the Notional Quantity schedule for a one or more groups of Settlement Periods to which the Notional Quantity is based. If the schedule
      differs for different groups of Settlement Periods, this element should be repeated.'

   End Choice
   Start Choice [1]
      <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

      <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

      <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityNotionalQuantitySchedule">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="notionalStep" type=" CommodityNotionalQuantity " maxOccurs="unbounded"/>
      <xsd:element name="settlementPeriodsNotionalQuantitySchedule" type=" CommoditySettlementPeriodsNotionalQuantitySchedule " maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityOption

[Table of contents]

Super-types:	Product < CommodityOption (by extension)
Sub-types:	None

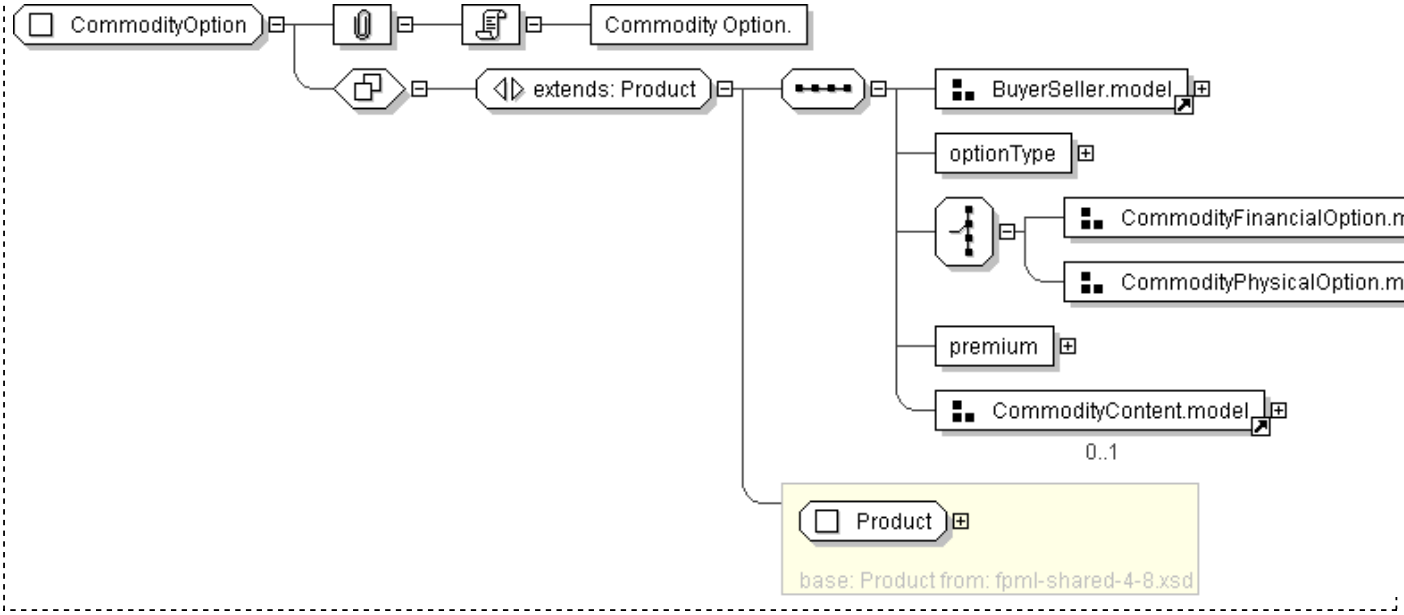
Name	CommodityOption
Used by (from the same schema document)	Element commodityOption
Abstract	no
Documentation	Commodity Option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization
   using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are not
   strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case
   of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the
   rights defined by this instrument and in return receives a payment for it. See 2000 ISDA
   definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> PutCallEnum </optionType> [1]
   'The type of option transaction.'

   Start Choice [1]
      <commodity> Commodity </commodity> [1]
      'Specifies the underlying component. At the time of the initial schema design, only
      underlyers of type Commodity are supported; the choice group in the future could offer
      the possibility of adding other types later.'

      Start Group: CommodityAsian.model [0..1]
'A group containing properties specific to Asian options.'


         <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
         'The effective date of the Commodity Option Transaction. Note that the
         Termination/Expiration Date should be specified in expirationDate within the
         CommodityAmericanExercise type or the CommodityEuropeanExercise type, as applicable.'

         Start Choice [1]
            <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule
            </calculationPeriodsSchedule> [1]
            'A parametric representation of the Calculation Periods of the Commodity Option
            Transaction.'

            <calculationPeriods> AdjustableDates </calculationPeriods> [1]
            'An absolute representation of the Calculation Period start dates of the Commodity
            Option Transaction.'

         End Choice
         <pricingDates> CommodityPricingDates </pricingDates> [1]
         'The dates on which the option will price.'
```

```
      <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
      'The Method of Averaging if there is more than one Pricing Date.'

    End Group: CommodityAsian.model
    Start Choice [1]
      Start Choice [1]
        <notionalQuantitySchedule> CommodityNotionalQuantitySchedule </notionalQuantitySchedule>
        [1]
        'Allows the documentation of a shaped notional trade where the notional changes
        over the life of the transaction.'

        <notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]
        'The Notional Quantity.'

        <settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity
        </settlementPeriodsNotionalQuantity> [1..*]
        'For an electricity transaction, the Notional Quantity for a one or more groups of
        Settlement Periods to which the Notional Quantity is based. If the schedule differs
        for different groups of Settlement Periods, this element should be repeated.'

      End Choice
      <totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]
      'The Total Notional Quantity.'

      <quantityReference> QuantityReference </quantityReference> [1]
      'A pointer style reference to a quantity defined on another leg.'

    End Choice
    <exercise> CommodityExercise </exercise> [1]
    'The parameters for defining how the commodity option can be exercised and how it is
    settled.'

    Start Choice [1]
      <strikePricePerUnit> NonNegativeMoney </strikePricePerUnit> [1]
      'The currency amount of the strike price per unit.'

      <strikePricePerUnitSchedule> CommodityStrikeSchedule </strikePricePerUnitSchedule> [1]
    End Choice
    Start Choice [1]
      <commoditySwap> ... </commoditySwap> [1]
      <commodityForward> ... </commodityForward> [1]
    End Choice
    <physicalExercise> CommodityPhysicalExercise </physicalExercise> [1]
    'The parameters for defining how the commodity option can be exercised into a physical
    transaction.'

  End Choice
  <premium> CommodityPremium </premium> [1]
  'The option premium payable by the buyer to the seller.'

  Start Group: CommodityContent.model [0..1]
    <commonPricing> xsd:boolean </commonPricing> [0..1]
    'Common pricing may be relevant for a Transaction that references more than one
    Commodity Reference Price. If Common Pricing is not specified as applicable, it will be
    deemed not to apply.'

    <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
    'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in ISDA
    2005 Commodity Definitions, as applicable.'

    <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption>
    [0..1]
    'The consequences of Bullion Settlement Disruption Events.'

    <rounding> Rounding </rounding> [0..1]
    'Rounding direction and precision for amounts.'

  End Group: CommodityContent.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityOption">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model "/>
        <xsd:element name="optionType" type=" PutCallEnum "/>
        <xsd:choice>
          <xsd:group ref=" CommodityFinancialOption.model "/>
          <xsd:group ref=" CommodityPhysicalOption.model "/>
        </xsd:choice>
        <xsd:element name="premium" type=" CommodityPremium "/>
        <xsd:group ref=" CommodityContent.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPayRelativeToEvent

[Table of contents]

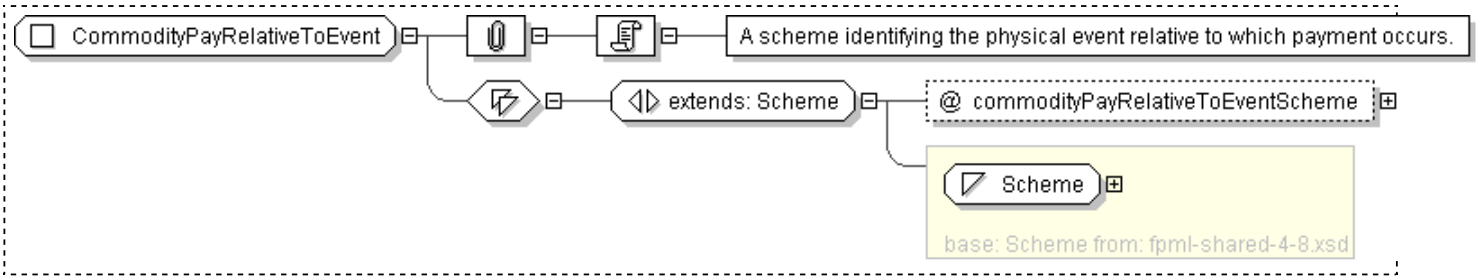
Super-types:	Scheme < CommodityPayRelativeToEvent (by extension)
Sub-types:	None

Name	CommodityPayRelativeToEvent
Used by (from the same schema document)	Complex Type CommodityRelativePaymentDates
Abstract	no
Documentation	A scheme identifying the physical event relative to which payment occurs.

XML Instance Representation

```
<...
 commodityPayRelativeToEventScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPayRelativeToEvent">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityPayRelativeToEventScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-pay-relative-to-event"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalAmericanExercise

[Table of contents]

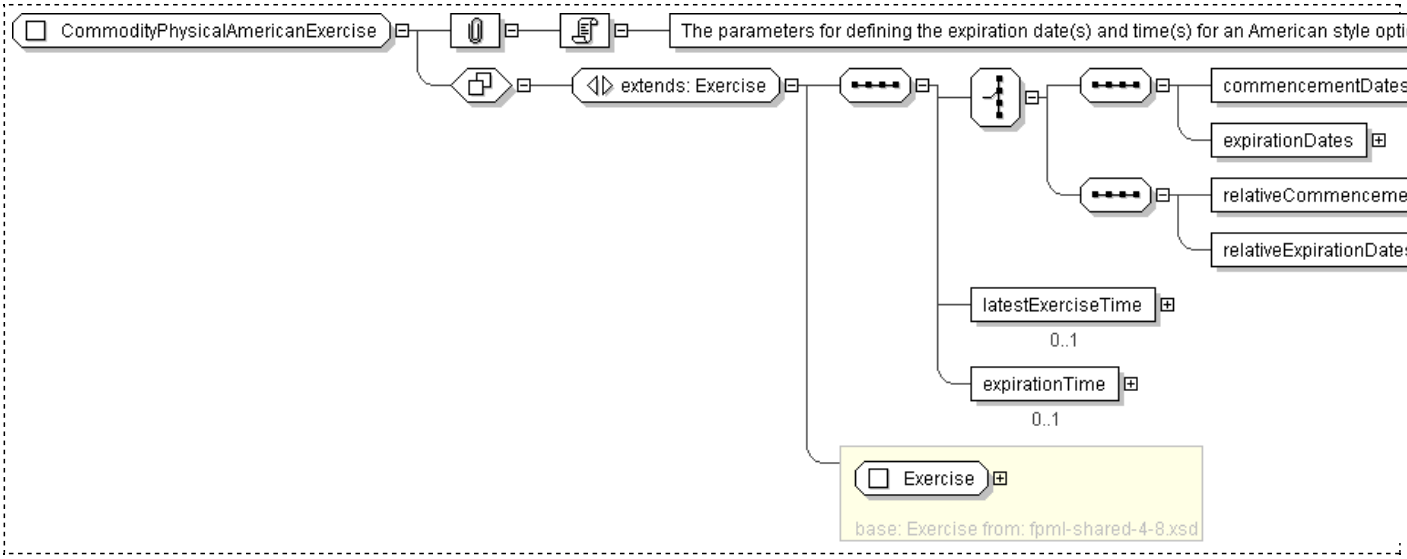
Super-types:	Exercise < CommodityPhysicalAmericanExercise (by extension)
Sub-types:	None

Name	CommodityPhysicalAmericanExercise
Used by (from the same schema document)	Complex Type CommodityPhysicalExercise
Abstract	no
Documentation	The parameters for defining the expiration date(s) and time(s) for an American style option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <commencementDates> AdjustableOrRelativeDates </commencementDates> [1]
      'The first day(s) of the exercise period(s) for an American-style option.'

      <expirationDates> AdjustableOrRelativeDates </expirationDates> [1]
      'The Expiration Date(s) of an American-style option.'

      <relativeCommencementDates> CommodityRelativeExpirationDates </relativeCommencementDates> [1]
      'The first day(s) of the exercise period(s) for an American-style option where it is relative to the
      occurrence of an external event.'

      <relativeExpirationDates> CommodityRelativeExpirationDates </relativeExpirationDates> [1]
      'The Expiration Date(s) of an American-style option where it is relative to the occurrence of an external
      event.'

   End Choice
   <latestExerciseTime> PrevailingTime </latestExerciseTime> [0..1]
   'For a Bermuda or American style option, the latest time on an exercise business day (excluding the
   expiration date) within the exercise period that notice can be given by the buyer to the seller or
   seller\'s agent. Notice of exercise given after this time will be deemed to have been given on the next
   exercise business day.'

   <expirationTime> PrevailingTime </expirationTime> [0..1]
   'The specific time of day at which the option expires.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalAmericanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:choice>
          <xsd:sequence>
            <xsd:element name="commencementDates" type=" AdjustableOrRelativeDates "/>
```

```
                <xsd:element name="expirationDates" type=" AdjustableOrRelativeDates "/>
              </xsd:sequence>
              <xsd:sequence>
                <xsd:element name="relativeCommencementDates" type=" CommodityRelativeExpirationDates "/>
                <xsd:element name="relativeExpirationDates" type=" CommodityRelativeExpirationDates "/>
              </xsd:sequence>
            </xsd:choice>
            <xsd:element name="latestExerciseTime" type=" PrevailingTime " minOccurs="0"/>
            <xsd:element name="expirationTime" type=" PrevailingTime " minOccurs="0"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalEuropeanExercise

[Table of contents]

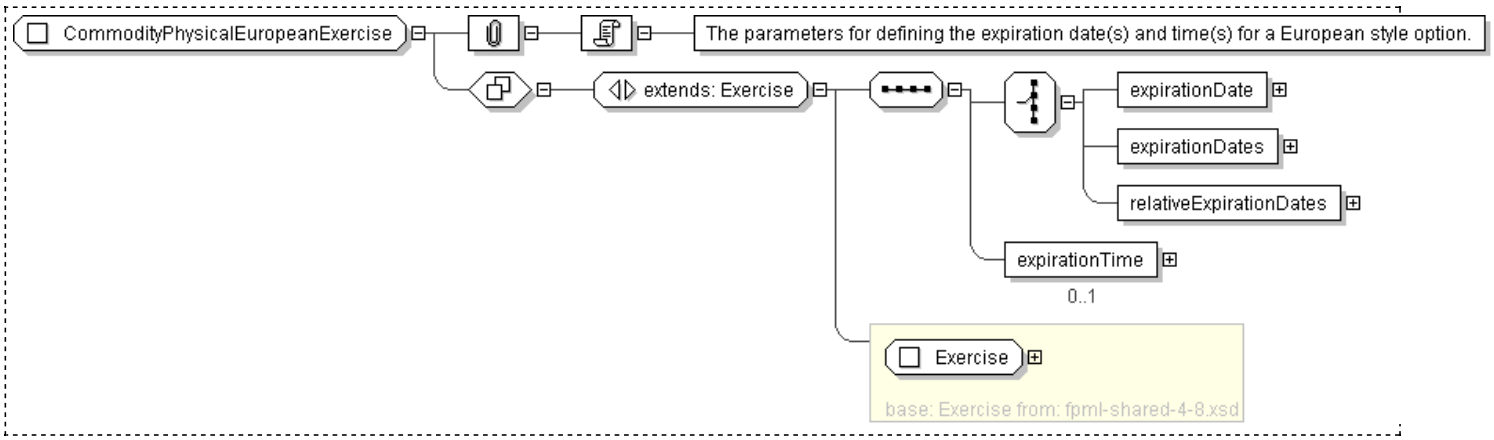
Super-types:	Exercise < CommodityPhysicalEuropeanExercise (by extension)
Sub-types:	None

Name	CommodityPhysicalEuropeanExercise
Used by (from the same schema document)	Complex Type CommodityPhysicalExercise
Abstract	no
Documentation	The parameters for defining the expiration date(s) and time(s) for a European style option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
      'The Expiration Date of a single expiry European-style option or the first Expiration Date of a
      multiple expiry or daily expiring option.'

      <expirationDates> AdjustableRelativeOrPeriodicDates2 </expirationDates> [1]
      'The Expiration Date(s) of a European-style option.'

      <relativeExpirationDates> CommodityRelativeExpirationDates </relativeExpirationDates> [1]
      'The Expiration Date(s) of a European-style option where it is relative to the occurrence of an
      external event.'

   End Choice
   <expirationTime> PrevailingTime </expirationTime> [0..1]
   'The specific time of day at which the option expires.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalEuropeanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
          <xsd:element name="expirationDates" type=" AdjustableRelativeOrPeriodicDates2 "/>
          <xsd:element name="relativeExpirationDates" type=" CommodityRelativeExpirationDates "/>
        </xsd:choice>
        <xsd:element name="expirationTime" type=" PrevailingTime " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalExercise

[Table of contents]

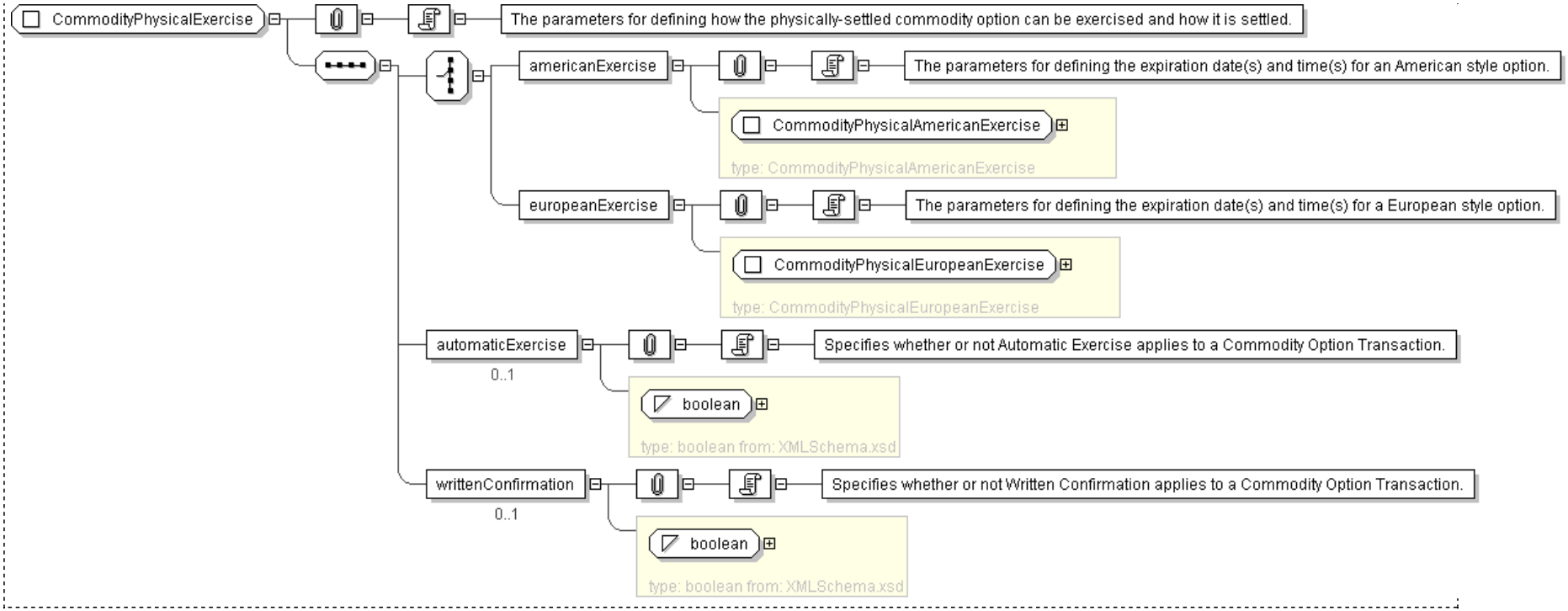
Super-types:	None
Sub-types:	None

Name	CommodityPhysicalExercise
Used by (from the same schema document)	Model Group CommodityPhysicalOption.model
Abstract	no
Documentation	The parameters for defining how the physically-settled commodity option can be exercised and how it is settled.

XML Instance Representation

```
<...>
  Start Choice [1]
    <americanExercise> CommodityPhysicalAmericanExercise </americanExercise> [1]
    'The parameters for defining the expiration date(s) and time(s) for an American style option.'

    <europeanExercise> CommodityPhysicalEuropeanExercise </europeanExercise> [1]
    'The parameters for defining the expiration date(s) and time(s) for a European style option.'

  End Choice
  <automaticExercise> xsd:boolean </automaticExercise> [0..1]
  'Specifies whether or not Automatic Exercise applies to a Commodity Option Transaction.'

  <writtenConfirmation> xsd:boolean </writtenConfirmation> [0..1]
  'Specifies whether or not Written Confirmation applies to a Commodity Option Transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalExercise">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="americanExercise" type=" CommodityPhysicalAmericanExercise "/>
      <xsd:element name="europeanExercise" type=" CommodityPhysicalEuropeanExercise "/>
    </xsd:choice>
    <xsd:element name="automaticExercise" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="writtenConfirmation" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalQuantity

[Table of contents]

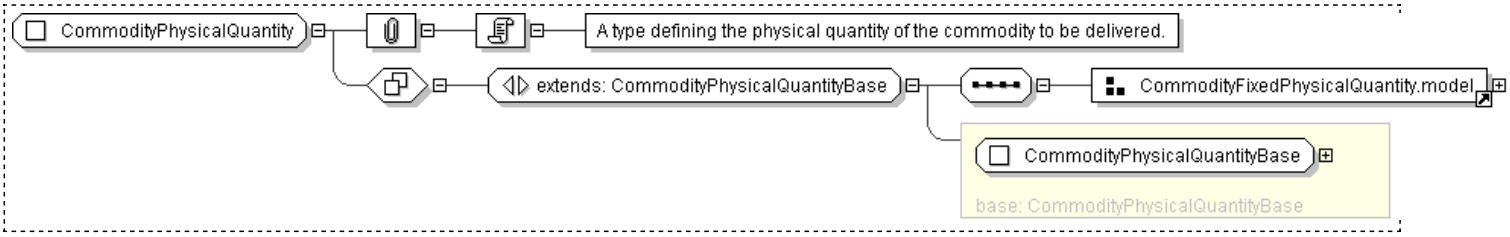
Super-types:	CommodityPhysicalQuantityBase < CommodityPhysicalQuantity (by extension)
Sub-types:	None

Name	CommodityPhysicalQuantity
Used by (from the same schema document)	Complex Type CoalPhysicalLeg , Complex Type OilPhysicalLeg
Abstract	no
Documentation	A type defining the physical quantity of the commodity to be delivered.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <physicalQuantity> CommodityNotionalQuantity </physicalQuantity> [1]
      'The Quantity per Delivery Period.'

      <physicalQuantitySchedule> CommodityPhysicalQuantitySchedule </physicalQuantitySchedule> [1]
      'Allows the documentation of a shaped quantity trade where the quantity changes over the life of the
      transaction.'

   End Choice
   <totalPhysicalQuantity> UnitQuantity </totalPhysicalQuantity> [0..1]
   'The Total Quantity of the commodity to be delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalQuantity">
  <xsd:complexContent>
    <xsd:extension base=" CommodityPhysicalQuantityBase ">
      <xsd:sequence>
        <xsd:group ref=" CommodityFixedPhysicalQuantity.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalQuantityBase

[Table of contents]

Super-types: None

Sub-types:

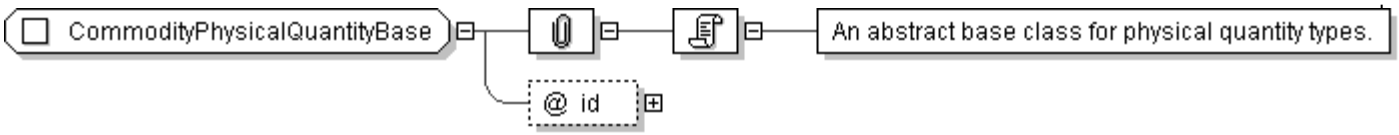
- CommodityPhysicalQuantity (by extension)
- ElectricityPhysicalQuantity (by extension)
- GasPhysicalQuantity (by extension)

Name	CommodityPhysicalQuantityBase
Abstract	yes
Documentation	An abstract base class for physical quantity types.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalQuantityBase" abstract="true">
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPhysicalQuantitySchedule

[Table of contents]

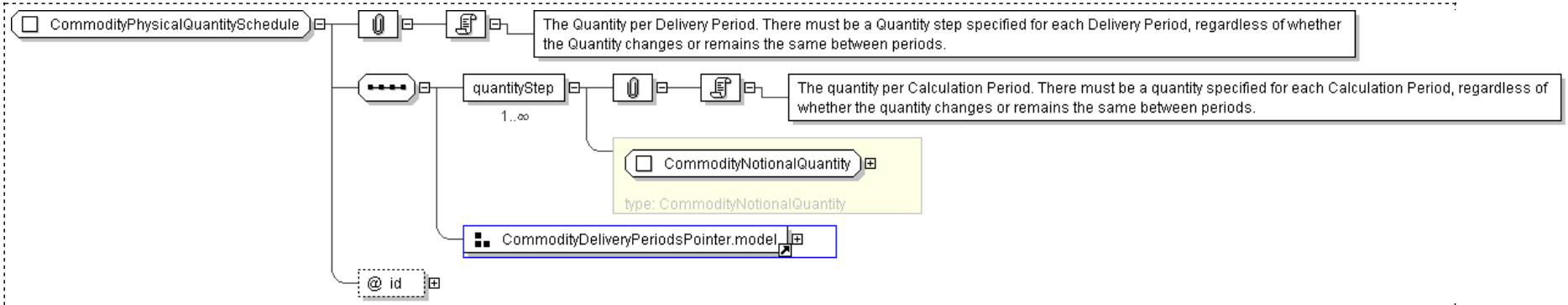
Super-types:	None
Sub-types:	• ElectricityPhysicalDeliveryQuantitySchedule (by extension)

Name	CommodityPhysicalQuantitySchedule
Used by (from the same schema document)	Model Group CommodityFixedPhysicalQuantity.model
Abstract	no
Documentation	The Quantity per Delivery Period. There must be a Quantity step specified for each Delivery Period, regardless of whether the Quantity changes or remains the same between periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityStep> CommodityNotionalQuantity </quantityStep> [1..*]
    'The quantity per Calculation Period. There must be a quantity specified for each Calculation Period, regardless of whether the quantity
    changes or remains the same between periods.'

   Start Choice [1]
      <deliveryPeriodsReference> CalculationPeriodsReference </deliveryPeriodsReference> [1]
       'A pointer style reference to the Delivery Periods defined elsewhere.'

      <deliveryPeriodsScheduleReference> CalculationPeriodsScheduleReference </deliveryPeriodsScheduleReference> [1]
       'A pointer style reference to the Calculation Periods Schedule defined elsewhere.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPhysicalQuantitySchedule">
  <xsd:sequence>
    <xsd:element name="quantityStep" type=" CommodityNotionalQuantity " maxOccurs="unbounded"/>
    <xsd:group ref=" CommodityDeliveryPeriodsPointer.model "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPipeline

[Table of contents]

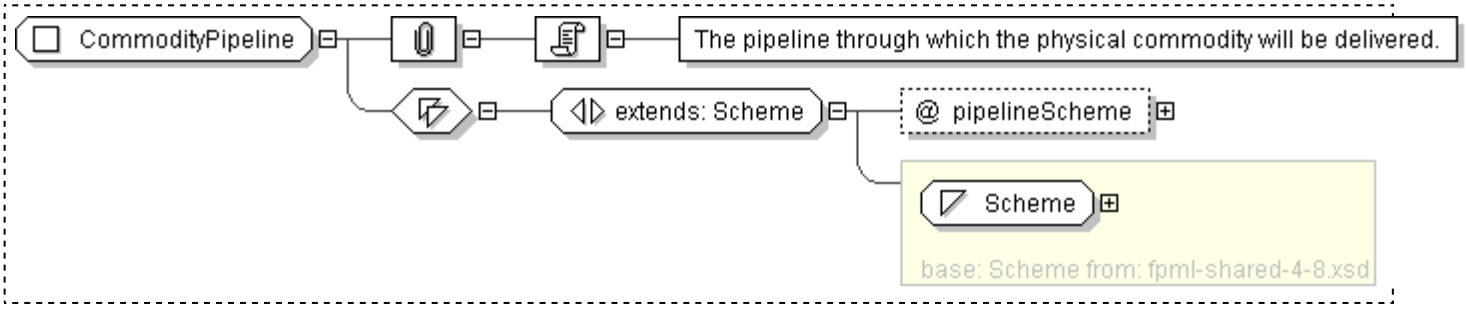
Super-types:	Scheme < CommodityPipeline (by extension)
Sub-types:	None

Name	CommodityPipeline
Used by (from the same schema document)	Complex Type OilPipelineDelivery
Abstract	no
Documentation	The pipeline through which the physical commodity will be delivered.

XML Instance Representation

```
<...
 pipelineScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPipeline">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="pipelineScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPipelineCycle

[Table of contents]

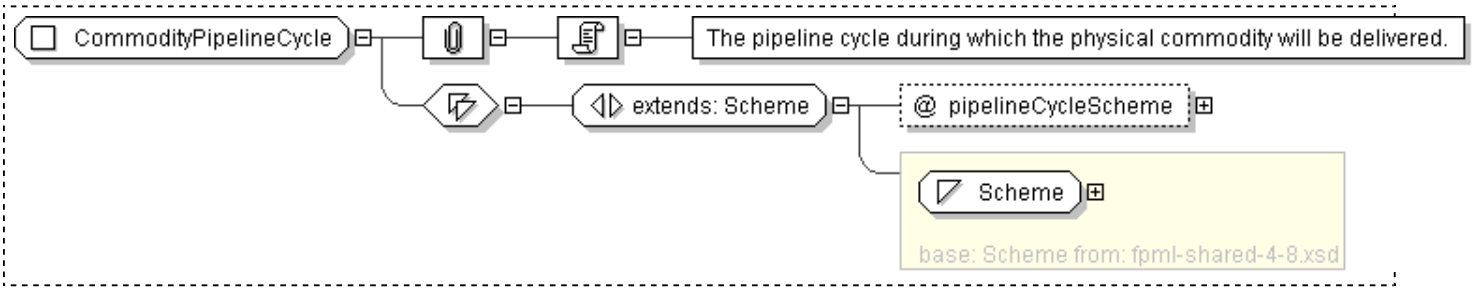
Super-types:	Scheme < CommodityPipelineCycle (by extension)
Sub-types:	None

Name	CommodityPipelineCycle
Used by (from the same schema document)	Complex Type OilPipelineDelivery
Abstract	no
Documentation	The pipeline cycle during which the physical commodity will be delivered.

XML Instance Representation

```
<...
 pipelineCycleScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPipelineCycle">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="pipelineCycleScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPremium

[Table of contents]

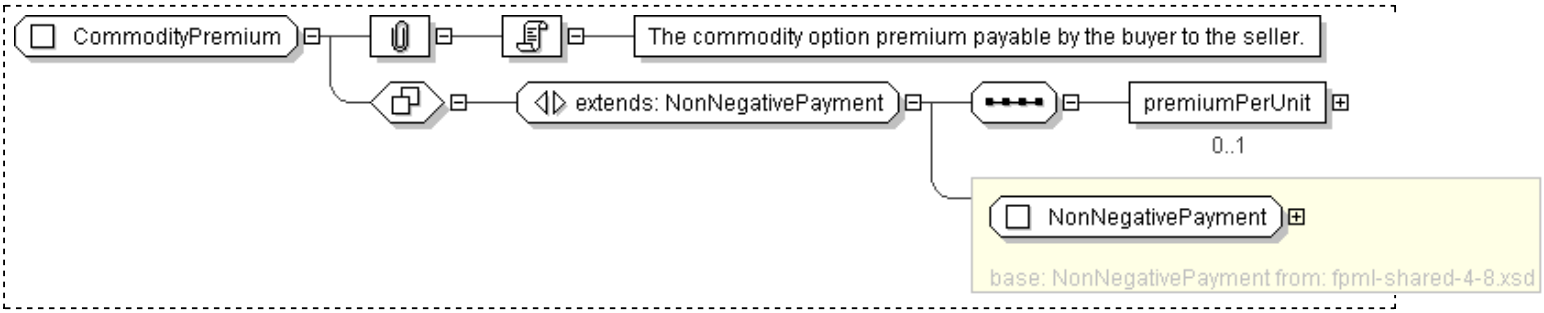
Super-types:	NonNegativePayment < CommodityPremium (by extension)
Sub-types:	None

Name	CommodityPremium
Used by (from the same schema document)	Complex Type CommodityOption
Abstract	no
Documentation	The commodity option premium payable by the buyer to the seller.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
   'The payment date, which can be expressed as either an adjustable or relative date.'

   <paymentAmount> NonNegativeMoney </paymentAmount> [1]
   'Non negative payment amount.'

   <premiumPerUnit> NonNegativeMoney </premiumPerUnit> [0..1]
   'The currency amount of premium to be paid per Unit of the Total Notional Quantity.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPremium">
   <xsd:complexContent>
      <xsd:extension base=" NonNegativePayment ">
         <xsd:sequence>
            <xsd:element name="premiumPerUnit" type=" NonNegativeMoney " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityPricingDates

[Table of contents]

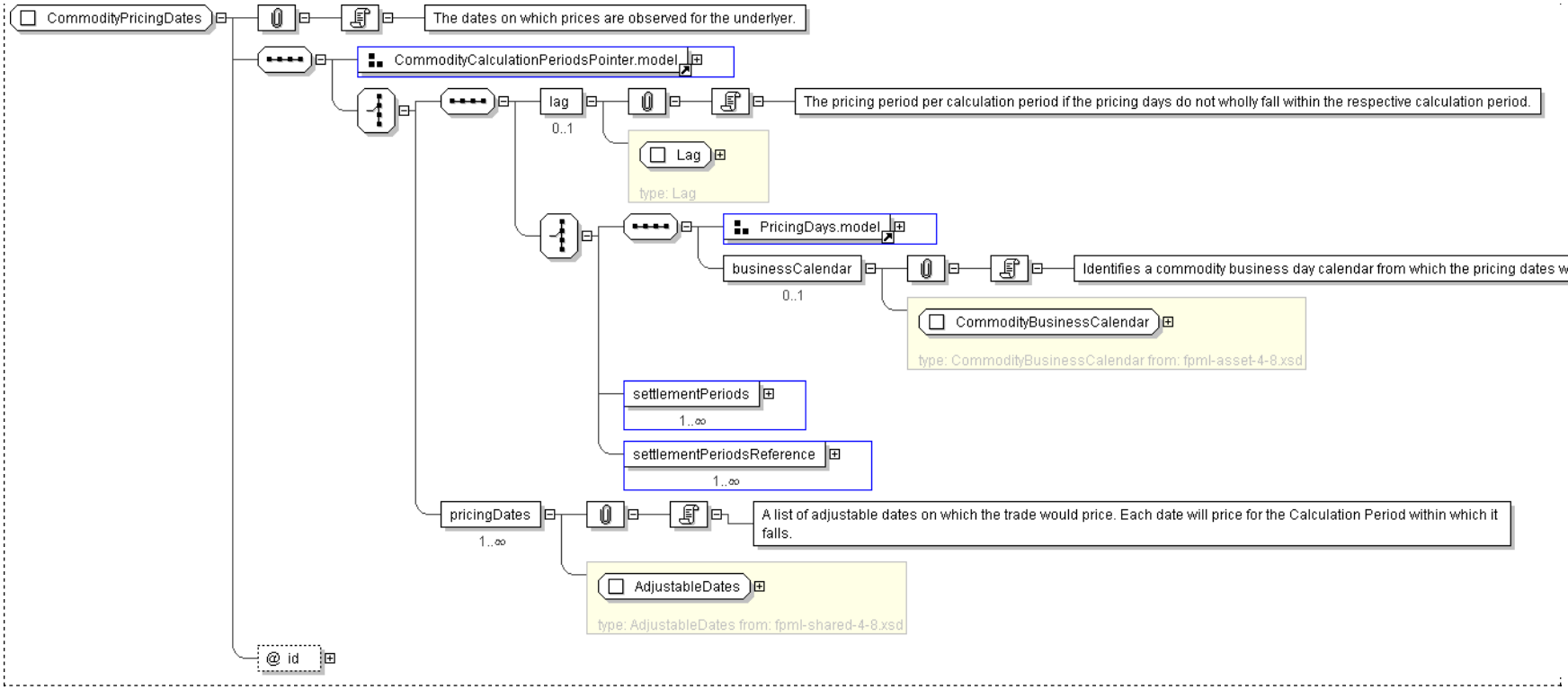
Super-types:	None
Sub-types:	None

Name	CommodityPricingDates
Used by (from the same schema document)	Complex Type FloatingLegCalculation , Model Group CommodityAsian.model
Abstract	no
Documentation	The dates on which prices are observed for the underlyer.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

      <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

      <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

   End Choice
   Start Choice [1]
      <lag> Lag </lag> [0..1]
      'The pricing period per calculation period if the pricing days do not wholly fall within the respective calculation period.'

      Start Choice [1]
         <dayType> CommodityDayTypeEnum </dayType> [1]
         'The type of day on which pricing occurs.'

         Start Choice [1]
            <dayDistribution> CommodityFrequencyType </dayDistribution> [1]
            'The method by which the pricing days are distributed across the pricing period.'

            <dayCount> xsd:positiveInteger </dayCount> [0..1]
            'The number of days over which pricing should take place.'

            <dayOfWeek> DayOfWeekEnum </dayOfWeek> [1..7]
            'The day(s) of the week on which pricing will take place during the pricing period.'

            <dayNumber> xsd:integer </dayNumber> [0..1]
            'The occurrence of the dayOfWeek within the pricing period on which pricing will take place, e.g. the 3rd Friday within each Calculation Period. If
            omitted, every dayOfWeek will be a pricing day.'

         End Choice
         <businessCalendar> CommodityBusinessCalendar </businessCalendar> [0..1]
         'Identifies a commodity business day calendar from which the pricing dates will be generated.'

         <settlementPeriods> SettlementPeriods </settlementPeriods> [1..*]
         <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
      End Choice
      <pricingDates> AdjustableDates </pricingDates> [1..*]
      'A list of adjustable dates on which the trade would price. Each date will price for the Calculation Period within which it falls.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityPricingDates">
  <xsd:sequence>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="lag" type=" Lag " minOccurs="0"/>
        <xsd:choice>
          <xsd:sequence>
            <xsd:group ref=" PricingDays.model "/>
            <xsd:element name="businessCalendar" type=" CommodityBusinessCalendar " minOccurs="0"/>
          </xsd:sequence>
          <xsd:element name="settlementPeriods" type=" SettlementPeriods " maxOccurs="unbounded"/>
          <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference " maxOccurs="unbounded"/>
        </xsd:choice>
      </xsd:sequence>
      <xsd:element name="pricingDates" type=" AdjustableDates " maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityProductGrade

[Table of contents]

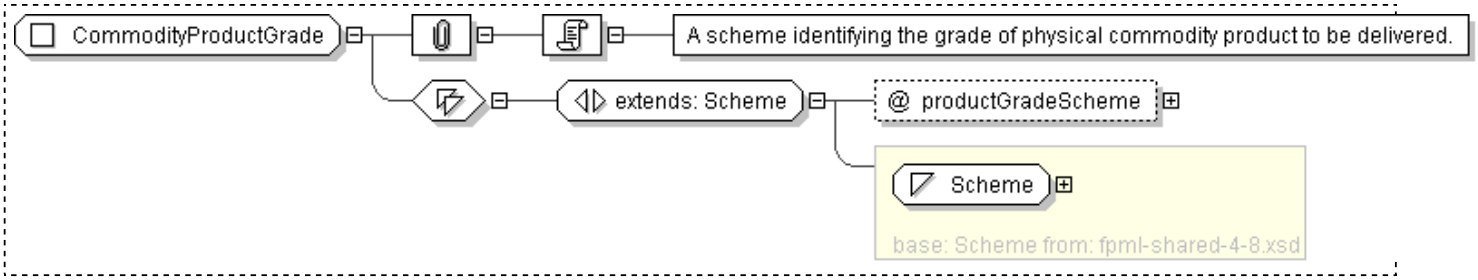
Super-types:	Scheme < CommodityProductGrade (by extension)
Sub-types:	None

Name	CommodityProductGrade
Used by (from the same schema document)	Complex Type OilProduct
Abstract	no
Documentation	A scheme identifying the grade of physical commodity product to be delivered.

XML Instance Representation

```
<...
 productGradeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityProductGrade">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="productGradeScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityQuantityFrequency

[Table of contents]

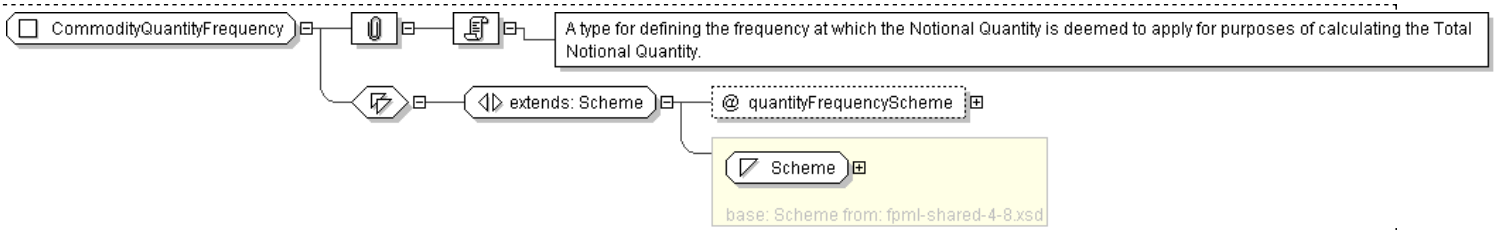
Super-types:	Scheme < CommodityQuantityFrequency (by extension)
Sub-types:	None

Name	CommodityQuantityFrequency
Used by (from the same schema document)	Complex Type CommodityNotionalQuantity
Abstract	no
Documentation	A type for defining the frequency at which the Notional Quantity is deemed to apply for purposes of calculating the Total Notional Quantity.

XML Instance Representation

```
<...
 quantityFrequencyScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityQuantityFrequency">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="quantityFrequencyScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/commodity-quantity-frequency"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityRelativeExpirationDates

[Table of contents]

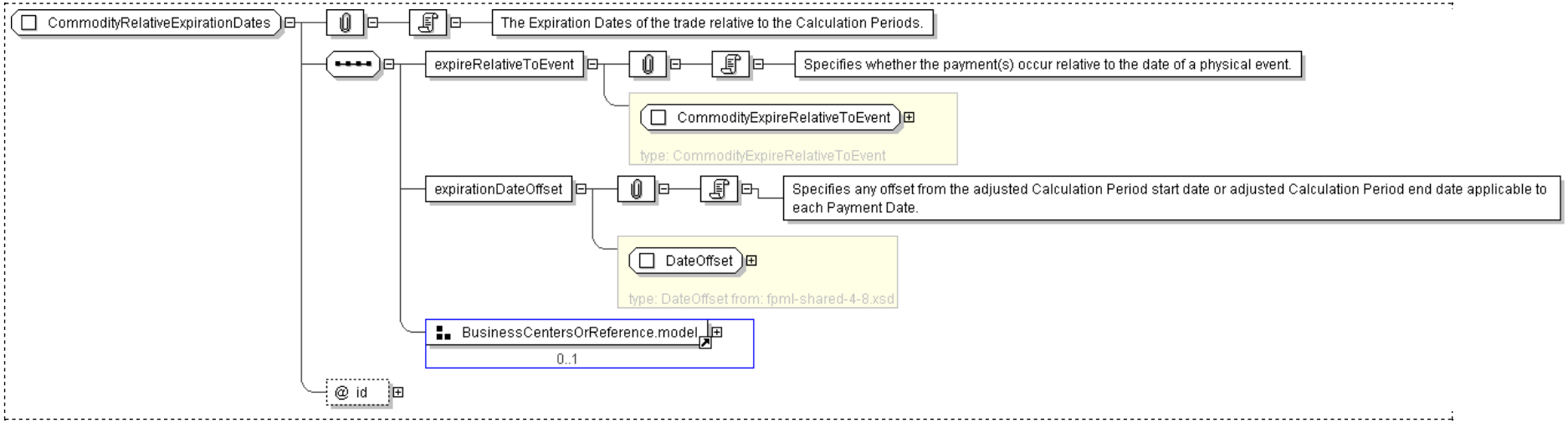
Super-types:	None
Sub-types:	None

Name	CommodityRelativeExpirationDates
Used by (from the same schema document)	Complex Type CommodityPhysicalAmericanExercise , Complex Type CommodityPhysicalAmericanExercise , Complex Type CommodityPhysicalEuropeanExercise
Abstract	no
Documentation	The Expiration Dates of the trade relative to the Calculation Periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <expireRelativeToEvent> CommodityExpireRelativeToEvent </expireRelativeToEvent> [1]
    'Specifies whether the payment(s) occur relative to the date of a physical event.'

   <expirationDateOffset> DateOffset </expirationDateOffset> [1]
    'Specifies any offset from the adjusted Calculation Period start date or adjusted Calculation Period end date applicable to each Payment Date.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
          'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to
          determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityRelativeExpirationDates">
  <xsd:sequence>
    <xsd:element name="expireRelativeToEvent" type=" CommodityExpireRelativeToEvent "/>
    <xsd:element name="expirationDateOffset" type=" DateOffset "/>
    <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
  </xsd:sequence>
```

```
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityRelativePaymentDates

[Table of contents]

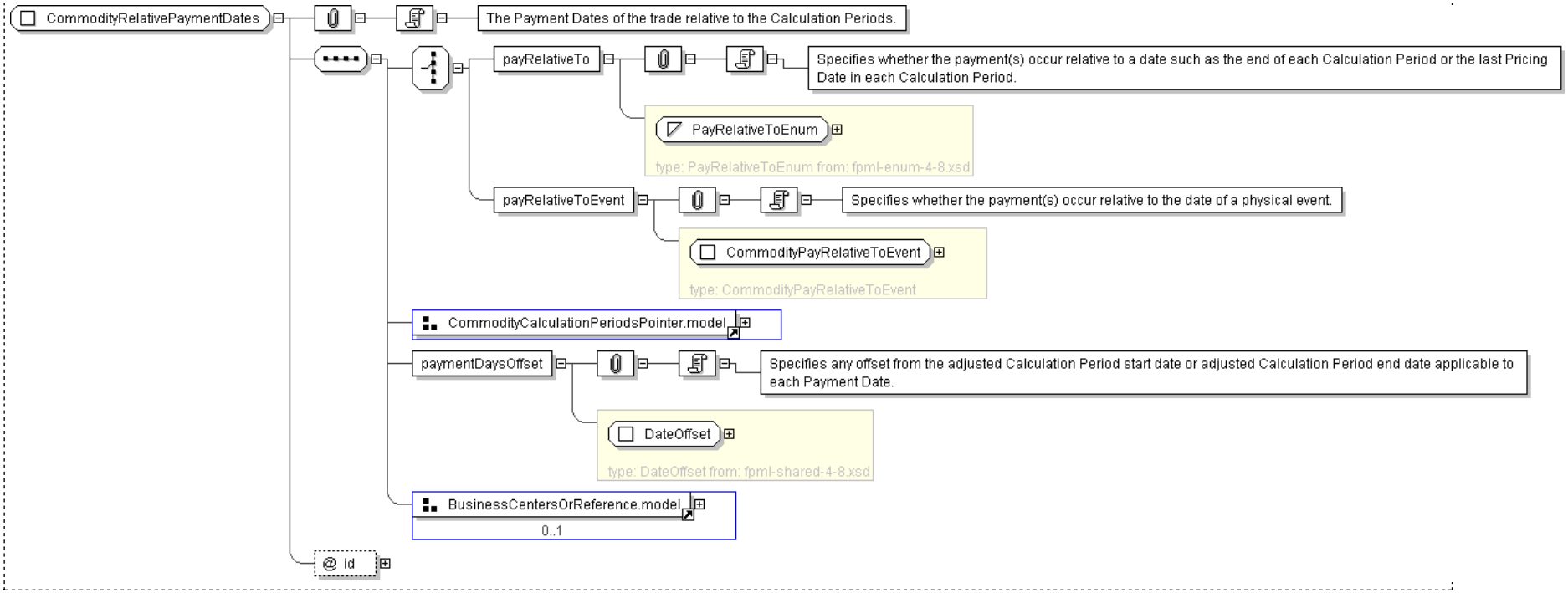
Super-types:	None
Sub-types:	None

Name	CommodityRelativePaymentDates
Used by (from the same schema document)	Model Group CommodityPaymentDates.model
Abstract	no
Documentation	The Payment Dates of the trade relative to the Calculation Periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <payRelativeTo> PayRelativeToEnum </payRelativeTo> [1]
      'Specifies whether the payment(s) occur relative to a date such as the end of each Calculation Period or the last Pricing Date in each Calculation
      Period.'

      <payRelativeToEvent> CommodityPayRelativeToEvent </payRelativeToEvent> [1]
      'Specifies whether the payment(s) occur relative to the date of a physical event.'

   End Choice
   Start Choice [1]
      <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

      <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

      <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

   End Choice
   <paymentDaysOffset> DateOffset </paymentDaysOffset> [1]
   'Specifies any offset from the adjusted Calculation Period start date or adjusted Calculation Period end date applicable to each Payment Date.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
         'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to
         determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityRelativePaymentDates">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="payRelativeTo" type=" PayRelativeToEnum "/>
      <xsd:element name="payRelativeToEvent" type=" CommodityPayRelativeToEvent "/>
    </xsd:choice>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
    <xsd:element name="paymentDaysOffset" type=" DateOffset "/>
    <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommoditySettlementPeriodsNotionalQuantity

[Table of contents]

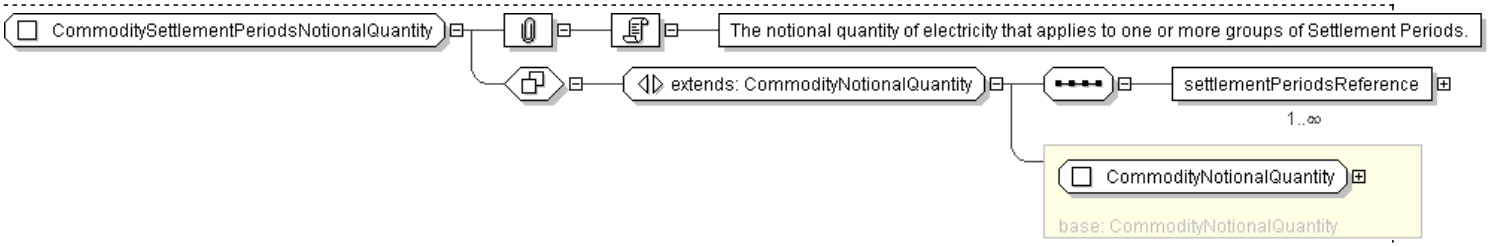
Super-types:	CommodityNotionalQuantity < CommoditySettlementPeriodsNotionalQuantity (by extension)
Sub-types:	None

Name	CommoditySettlementPeriodsNotionalQuantity
Used by (from the same schema document)	Model Group CommodityNotionalQuantity.model
Abstract	no
Documentation	The notional quantity of electricity that applies to one or more groups of Settlement Periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityUnit> QuantityUnit </quantityUnit> [1]
   'Quantity Unit is the unit of measure applicable for the quantity on the Transaction.'

   <quantityFrequency> CommodityQuantityFrequency </quantityFrequency> [1]
   'The frequency at which the Notional Quantity is deemed to apply for purposes of calculating the Total
   Notional Quantity.'

   <quantity> xsd:decimal </quantity> [1]
   'Amount of commodity per quantity frequency.'

   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
   'The range(s) of Settlement Periods to which the Notional Quantity applies.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommoditySettlementPeriodsNotionalQuantity">
  <xsd:complexContent>
    <xsd:extension base=" CommodityNotionalQuantity ">
      <xsd:sequence>
        <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommoditySettlementPeriodsNotionalQuantitySchedule

[Table of contents]

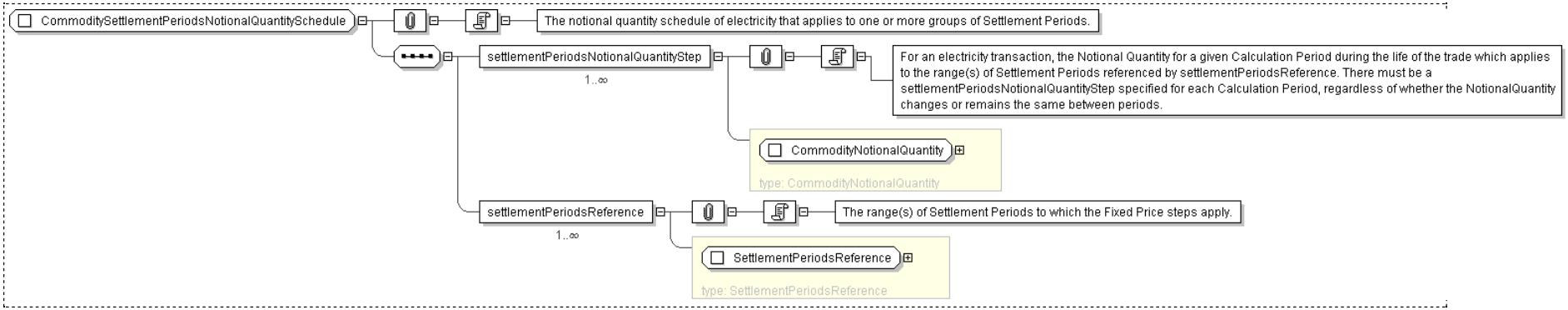
Super-types:	None
Sub-types:	None

Name	CommoditySettlementPeriodsNotionalQuantitySchedule
Used by (from the same schema document)	Complex Type CommodityNotionalQuantitySchedule
Abstract	no
Documentation	The notional quantity schedule of electricity that applies to one or more groups of Settlement Periods.

XML Instance Representation

```
<...>
   <settlementPeriodsNotionalQuantityStep> CommodityNotionalQuantity </settlementPeriodsNotionalQuantityStep> [1..*]
   'For an electricity transaction, the Notional Quantity for a given Calculation Period during the life of the trade which applies to the range(s) of Settlement Periods
   referenced by settlementPeriodsReference. There must be a settlementPeriodsNotionalQuantityStep specified for each Calculation Period, regardless of whether the
   NotionalQuantity changes or remains the same between periods.'

   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
   'The range(s) of Settlement Periods to which the Fixed Price steps apply.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommoditySettlementPeriodsNotionalQuantitySchedule">
  <xsd:sequence>
    <xsd:element name="settlementPeriodsNotionalQuantityStep" type=" CommodityNotionalQuantity " maxOccurs="unbounded"/>
    <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommoditySettlementPeriodsPriceSchedule

[Table of contents]

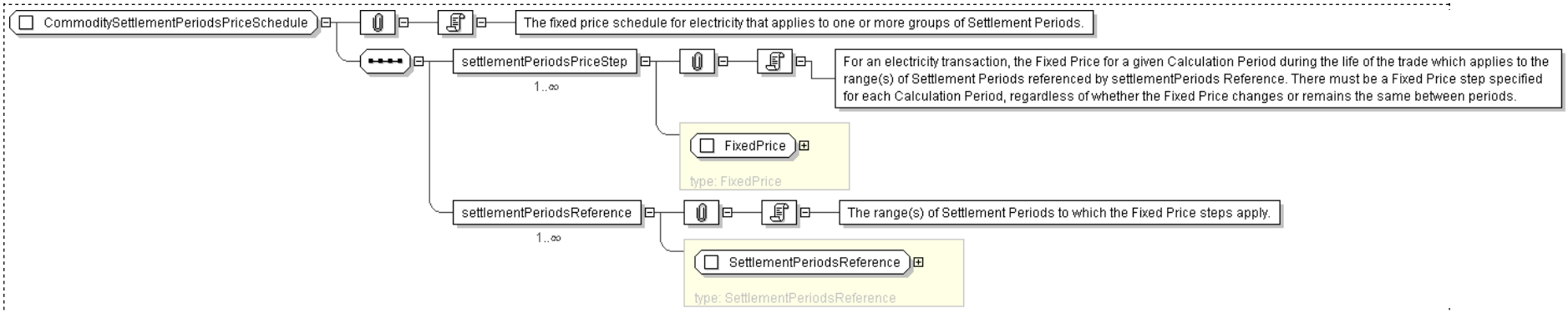
Super-types:	None
Sub-types:	None

Name	CommoditySettlementPeriodsPriceSchedule
Used by (from the same schema document)	Complex Type CommodityFixedPriceSchedule
Abstract	no
Documentation	The fixed price schedule for electricity that applies to one or more groups of Settlement Periods.

XML Instance Representation

```
<...>
   <settlementPeriodsPriceStep> FixedPrice </settlementPeriodsPriceStep> [1..*]
   'For an electricity transaction, the Fixed Price for a given Calculation Period during the life of the trade which applies to the range(s) of Settlement Periods
   referenced by settlementPeriods Reference. There must be a Fixed Price step specified for each Calculation Period, regardless of whether the Fixed Price changes
   or remains the same between periods.'

   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
   'The range(s) of Settlement Periods to which the Fixed Price steps apply.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommoditySettlementPeriodsPriceSchedule">
   <xsd:sequence>
      <xsd:element name="settlementPeriodsPriceStep" type=" FixedPrice " maxOccurs="unbounded"/>
      <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommoditySpreadSchedule

[Table of contents]

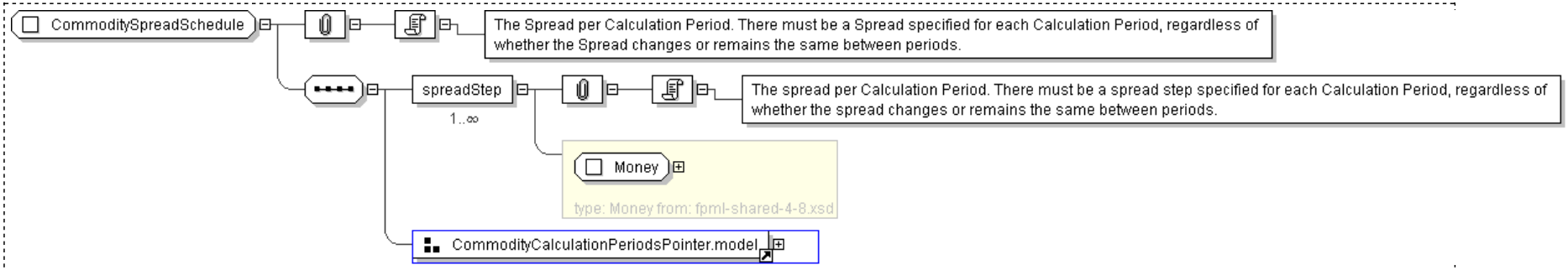
Super-types:	None
Sub-types:	None

Name	CommoditySpreadSchedule
Used by (from the same schema document)	Complex Type FloatingLegCalculation
Abstract	no
Documentation	The Spread per Calculation Period. There must be a Spread specified for each Calculation Period, regardless of whether the Spread changes or remains the same between periods.

XML Instance Representation

```
<...>
  <spreadStep> Money </spreadStep> [1..*]
  'The spread per Calculation Period. There must be a spread step specified for each Calculation Period, regardless of whether the spread
  changes or remains the same between periods.'

  Start Choice [1]
    <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
    'A pointer style reference to the Calculation Periods defined on another leg.'

    <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
    'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

    <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
    'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommoditySpreadSchedule">
  <xsd:sequence>
    <xsd:element name="spreadStep" type=" Money " maxOccurs="unbounded"/>
    <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommodityStrikeSchedule

[Table of contents]

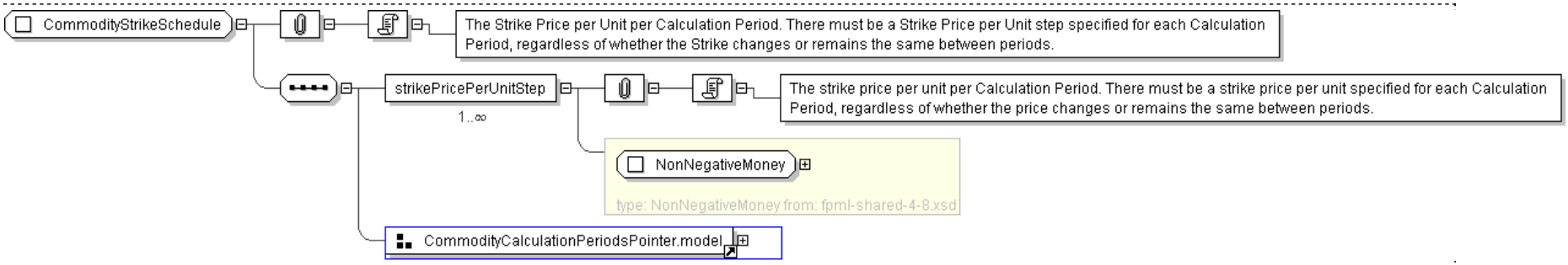
Super-types:	None
Sub-types:	None

Name	CommodityStrikeSchedule
Used by (from the same schema document)	Model Group CommodityStrikePrice.model
Abstract	no
Documentation	The Strike Price per Unit per Calculation Period. There must be a Strike Price per Unit step specified for each Calculation Period, regardless of whether the Strike changes or remains the same between periods.

XML Instance Representation

```
<...>
   <strikePricePerUnitStep> NonNegativeMoney </strikePricePerUnitStep> [1..*]
   'The strike price per unit per Calculation Period. There must be a strike price per unit specified for each Calculation Period, regardless of
   whether the price changes or remains the same between periods.'

   Start Choice [1]
      <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
      'A pointer style reference to the Calculation Periods defined on another leg.'

      <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference </calculationPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

      <calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]
      'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommodityStrikeSchedule">
   <xsd:sequence>
      <xsd:element name="strikePricePerUnitStep" type=" NonNegativeMoney " maxOccurs="unbounded"/>
      <xsd:group ref=" CommodityCalculationPeriodsPointer.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CommoditySwap

[Table of contents]

Super-types:	Product < CommoditySwap (by extension)
Sub-types:	None

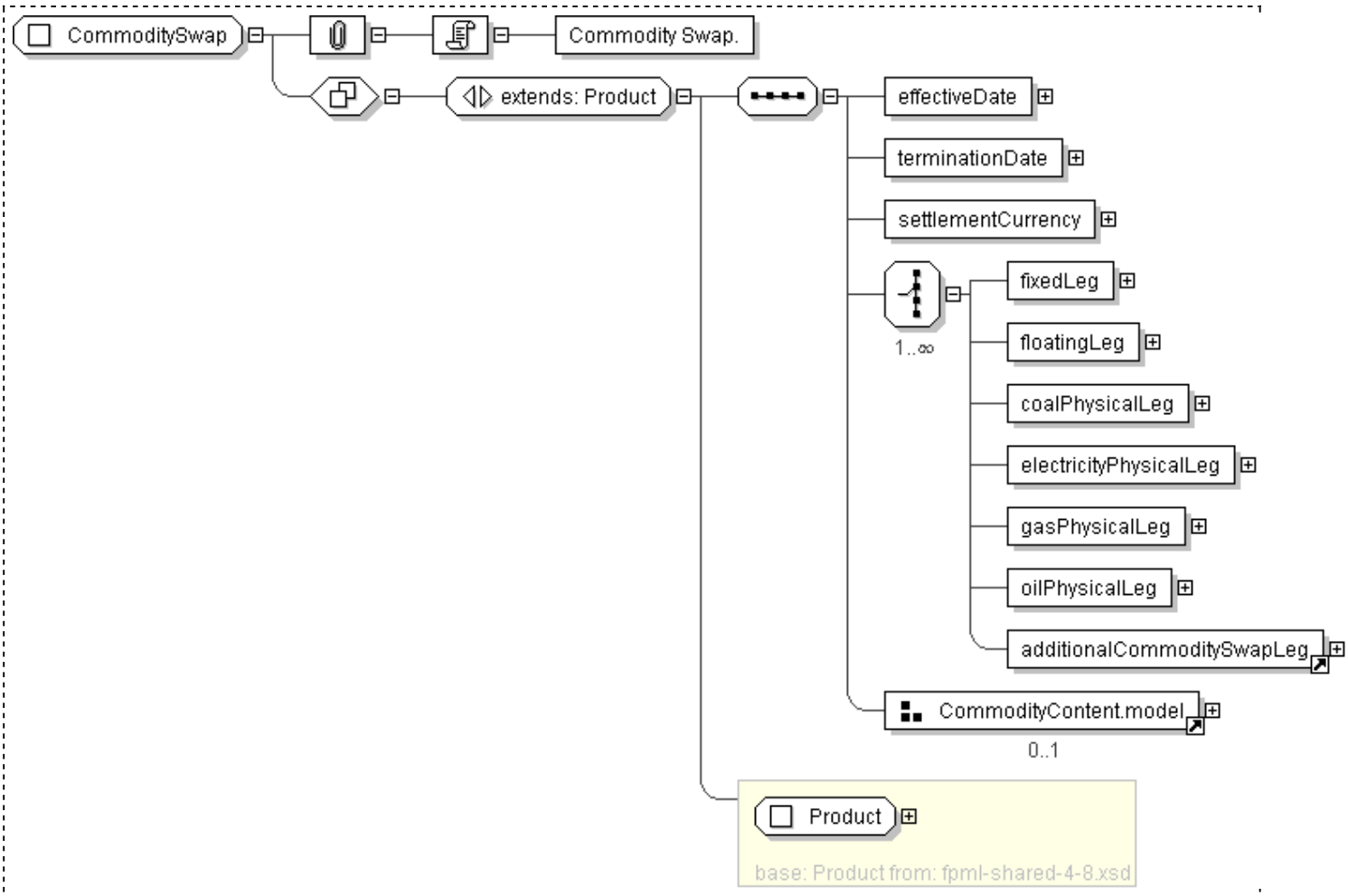
Name	CommoditySwap
Used by (from the same schema document)	Element commoditySwap
Abstract	no
Documentation	Commodity Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component),
   this element will typically point to the effective date of the other leg of the
   swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of this leg of the swap. When defined in relation to
   a date specified somewhere else in the document (through the relativeDate component),
   this element will typically point to the termination date of the other leg of the
   swap.'

   <settlementCurrency> IdentifiedCurrency </settlementCurrency> [1]
   'The currency into which the Commodity Swap Transaction will settle. If this is not
   the same as the currency in which the Commodity Reference Price is quoted on a given
   floating leg of the Commodity Swap Transaction, then an FX rate should also be
   specified for that leg.'

   Start Choice [1..*]
     <fixedLeg> FixedPriceLeg </fixedLeg> [1]
     'Fixed Price Leg.'

     <floatingLeg> FloatingPriceLeg </floatingLeg> [1]
     'Floating Price leg.'

     <coalPhysicalLeg> CoalPhysicalLeg </coalPhysicalLeg> [1]
     'Physically settled coal leg.'

     <electricityPhysicalLeg> ElectricityPhysicalLeg </electricityPhysicalLeg> [1]
     'Physically settled electricity leg.'

     <gasPhysicalLeg> GasPhysicalLeg </gasPhysicalLeg> [1]
     'Physically settled natural gas leg.'

     <oilPhysicalLeg> OilPhysicalLeg </oilPhysicalLeg> [1]
     'Physically settled oil or refined products leg.'
```

```
      <additionalCommoditySwapLeg> ... </additionalCommoditySwapLeg> [1]
   End Choice
   Start Group: CommodityContent.model [0..1]
      <commonPricing> xsd:boolean </commonPricing> [0..1]
      'Common pricing may be relevant for a Transaction that references more than one
      Commodity Reference Price. If Common Pricing is not specified as applicable, it
      will be deemed not to apply.'

      <marketDisruption> CommodityMarketDisruption </marketDisruption> [0..1]
      'Market disruption events as defined in the ISDA 1993 Commodity Definitions or in
      ISDA 2005 Commodity Definitions, as applicable.'

      <settlementDisruption> CommodityBullionSettlementDisruptionEnum </settlementDisruption>
      [0..1]
      'The consequences of Bullion Settlement Disruption Events.'

      <rounding> Rounding </rounding> [0..1]
      'Rounding direction and precision for amounts.'

   End Group: CommodityContent.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CommoditySwap">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="settlementCurrency" type=" IdentifiedCurrency "/>
            <xsd:choice maxOccurs="unbounded">
               <xsd:element name="fixedLeg" type=" FixedPriceLeg "/>
               <xsd:element name="floatingLeg" type=" FloatingPriceLeg "/>
               <xsd:element name="coalPhysicalLeg" type=" CoalPhysicalLeg "/>
```

```
            <xsd:element name="electricityPhysicalLeg" type=" ElectricityPhysicalLeg "/>
            <xsd:element name="gasPhysicalLeg" type=" GasPhysicalLeg "/>
            <xsd:element name="oilPhysicalLeg" type=" OilPhysicalLeg "/>
            <xsd:element ref=" additionalCommoditySwapLeg "/>
          </xsd:choice>
          <xsd:group ref=" CommodityContent.model " minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DisruptionFallback

[Table of contents]

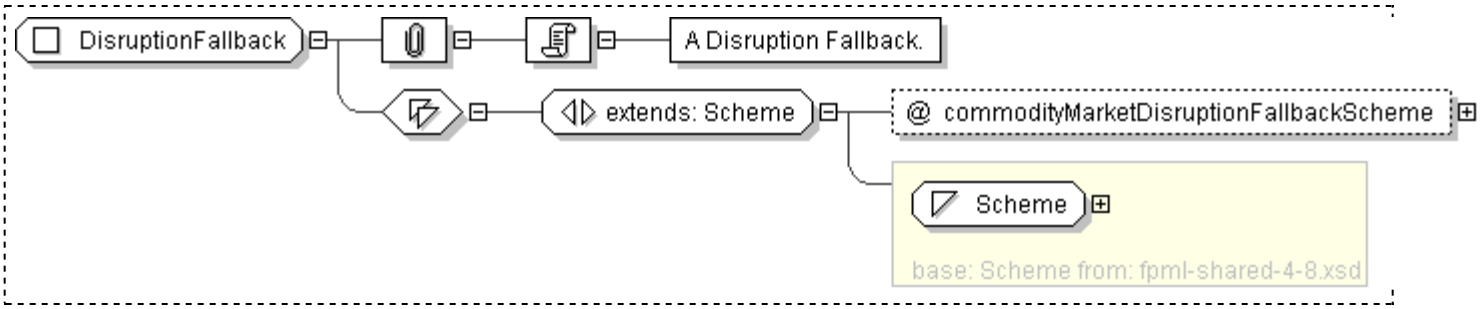
Super-types:	Scheme < DisruptionFallback (by extension)
Sub-types:	None

Name	DisruptionFallback
Used by (from the same schema document)	Complex Type SequencedDisruptionFallback
Abstract	no
Documentation	A Disruption Fallback.

XML Instance Representation

```
<...
 commodityMarketDisruptionFallbackScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DisruptionFallback">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityMarketDisruptionFallbackScheme" type="
         xsd:anyURI " default="http://www.fpml.org/coding-scheme/commodity-market-
         disruption-fallback"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDelivery

[Table of contents]

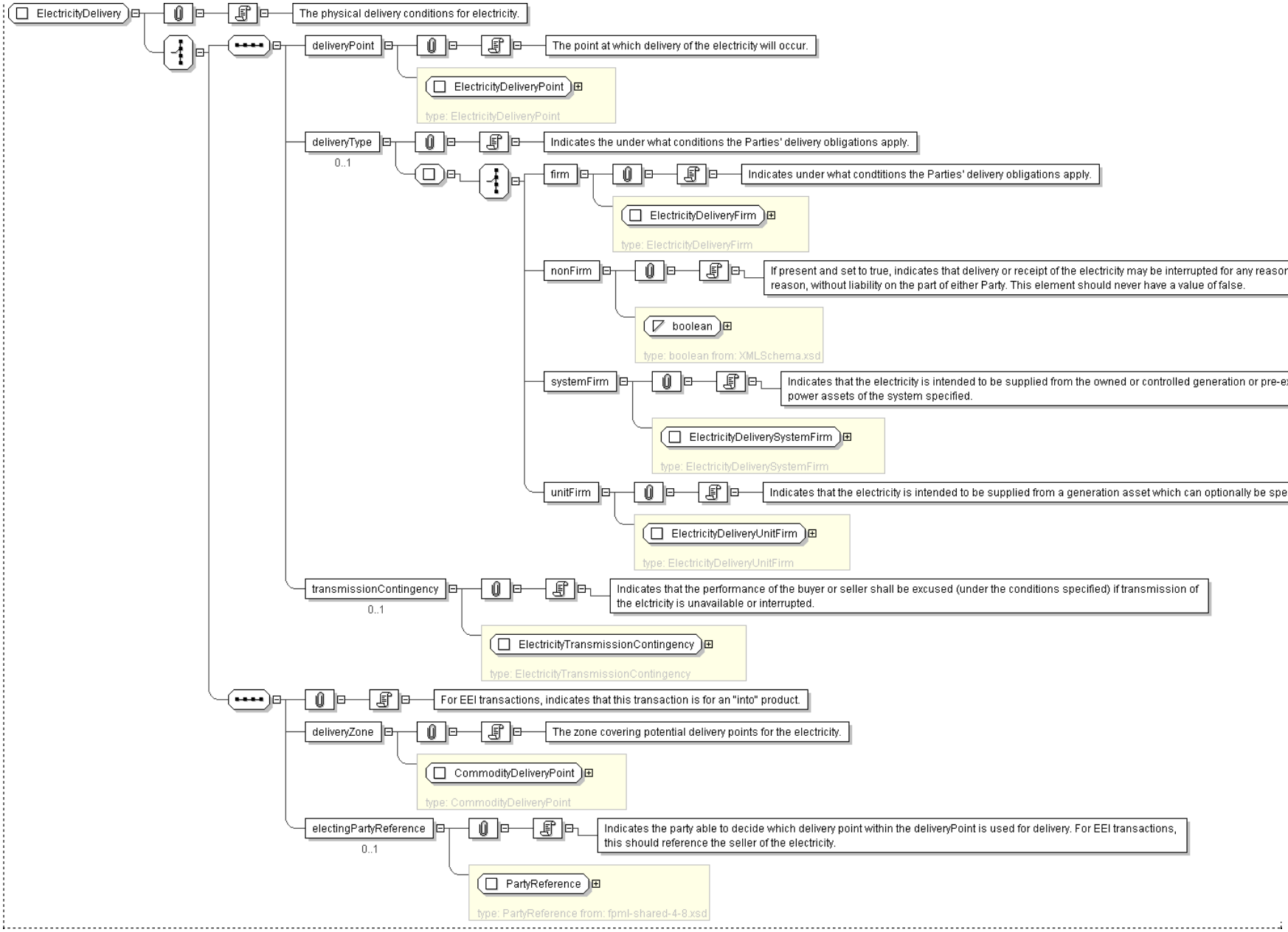
Super-types:	None
Sub-types:	None

Name	ElectricityDelivery
Used by (from the same schema document)	Complex Type ElectricityPhysicalLeg
Abstract	no
Documentation	The physical delivery conditions for electricity.

XML Instance Representation

```
<...>
  Start Choice [1]
    <deliveryPoint> ElectricityDeliveryPoint </deliveryPoint> [1]
    'The point at which delivery of the electricity will occur.'

    <deliveryType> [0..1]
    'Indicates the under what conditions the Parties\' delivery obligations apply.'


      Start Choice [1]
        <firm> ElectricityDeliveryFirm </firm> [1]
        'Indicates under what condtitions the Parties\' delivery obligations apply.'

        <nonFirm> xsd:boolean </nonFirm> [1]
        'If present and set to true, indicates that delivery or receipt of the electricity may be interrupted for any reason or for no reason, without liability on the part
        of either Party. This element should never have a value of false.'

        <systemFirm> ElectricityDeliverySystemFirm </systemFirm> [1]
        'Indicates that the electricity is intended to be supplied from the owned or controlled generation or pre-existing purchased power assets of the system specified.'

        <unitFirm> ElectricityDeliveryUnitFirm </unitFirm> [1]
        'Indicates that the electricity is intended to be supplied from a generation asset which can optionally be specified.'

      End Choice
    </deliveryType>
    <transmissionContingency> ElectricityTransmissionContingency </transmissionContingency> [0..1]
    'Indicates that the performance of the buyer or seller shall be excused (under the conditions specified) if transmission of the elctricity is unavailable or interrupted.'

    <deliveryZone> CommodityDeliveryPoint </deliveryZone> [1]
    'The zone covering potential delivery points for the electricity.'

    <electingPartyReference> PartyReference </electingPartyReference> [0..1]
    'Indicates the party able to decide which delivery point within the deliveryPoint is used for delivery. For EEI transactions, this should reference the seller of the
    electricity.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDelivery">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="deliveryPoint" type=" ElectricityDeliveryPoint "/>
      <xsd:element name="deliveryType" minOccurs="0">
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="firm" type=" ElectricityDeliveryFirm "/>
            <xsd:element name="nonFirm" type=" xsd:boolean "/>
            <xsd:element name="systemFirm" type=" ElectricityDeliverySystemFirm "/>
            <xsd:element name="unitFirm" type=" ElectricityDeliveryUnitFirm "/>
          </xsd:choice>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="transmissionContingency" type=" ElectricityTransmissionContingency " minOccurs="0"/>
```

```
      </xsd:sequence>
      <xsd:sequence>
         <xsd:element name="deliveryZone" type=" CommodityDeliveryPoint "/>
         <xsd:element name="electingPartyReference" type=" PartyReference " minOccurs="0"/>
      </xsd:sequence>
   </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDeliveryFirm

[Table of contents]

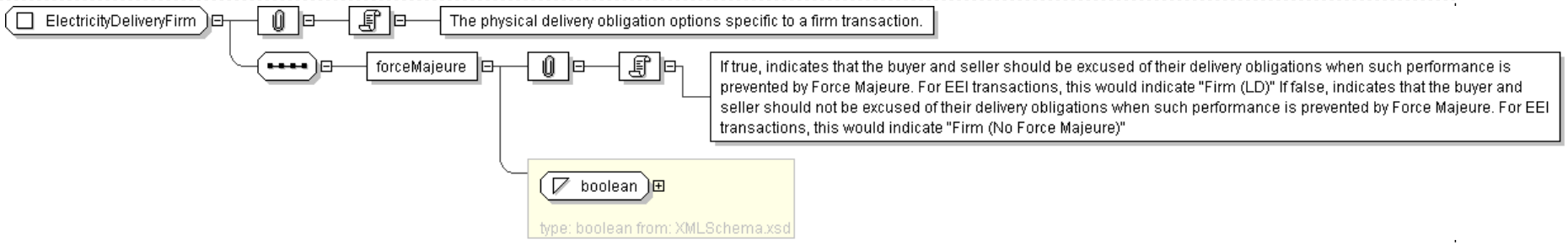
Super-types:	None
Sub-types:	None

Name	ElectricityDeliveryFirm
Used by (from the same schema document)	Complex Type ElectricityDelivery
Abstract	no
Documentation	The physical delivery obligation options specific to a firm transaction.

XML Instance Representation

```
<...>
  <forceMajeure> xsd:boolean </forceMajeure> [1]
   'If true, indicates that the buyer and seller should be excused of their delivery obligations when such performance is prevented by Force
   Majeure. For EEI transactions, this would indicate \"Firm (LD)\" If false, indicates that the buyer and seller should not be excused of
   their delivery obligations when such performance is prevented by Force Majeure. For EEI transactions, this would indicate \"Firm (No
   Force Majeure)\"'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDeliveryFirm">
  <xsd:sequence>
    <xsd:element name="forceMajeure" type=" xsd:boolean "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDeliveryPeriods

[Table of contents]

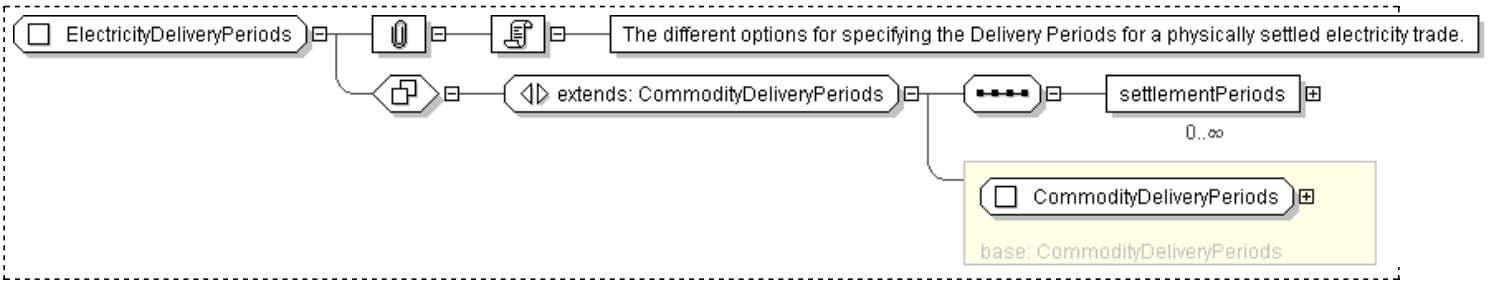
Super-types:	CommodityDeliveryPeriods < ElectricityDeliveryPeriods (by extension)
Sub-types:	None

Name	ElectricityDeliveryPeriods
Abstract	no
Documentation	The different options for specifying the Delivery Periods for a physically settled electricity trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <periods> AdjustableDates </periods> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if the
      Delivery Periods differ from the Calculation Periods on the fixed or floating leg. If
      DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be used
      to point to the Calculation Periods on that leg - or the calculationPeriodsScheduleReference can
      be used to point to the Calculation Periods Schedule for that leg.'

      <periodsSchedule> CommodityCalculationPeriodsSchedule </periodsSchedule> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if the
      Delivery Periods differ from the Calculation Periods on the fixed or floating leg. If
      DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be used
      to point to the Calculation Periods on that leg - or the calculationPeriodsScheduleReference can
      be used to point to the Calculation Periods Schedule for that leg.'

      Start Choice [1]
         <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
         'A pointer style reference to the Calculation Periods defined on another leg.'

         <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference
         </calculationPeriodsScheduleReference> [1]
         'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

         <calculationPeriodsDatesReference> CalculationPeriodsDatesReference
         </calculationPeriodsDatesReference> [1]
         'A pointer style reference to single-day-duration Calculation Periods defined on another
         leg.'

      End Choice
   End Choice
   <settlementPeriods> SettlementPeriods </settlementPeriods> [0..*]
   'The periods within the Delivery Periods during which the electricity will be delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDeliveryPeriods">
  <xsd:complexContent>
    <xsd:extension base=" CommodityDeliveryPeriods ">
      <xsd:sequence>
        <xsd:element name="settlementPeriods" type=" SettlementPeriods " minOccurs="0"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDeliveryPoint

[Table of contents]

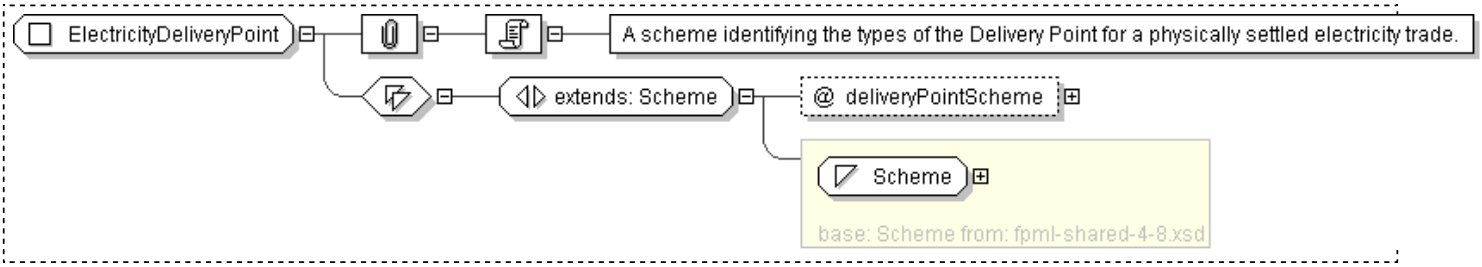
Super-types:	Scheme < ElectricityDeliveryPoint (by extension)
Sub-types:	None

Name	ElectricityDeliveryPoint
Used by (from the same schema document)	Complex Type ElectricityDelivery
Abstract	no
Documentation	A scheme identifying the types of the Delivery Point for a physically settled electricity trade.

XML Instance Representation

```
<...
 deliveryPointScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDeliveryPoint">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="deliveryPointScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDeliverySystemFirm

[Table of contents]

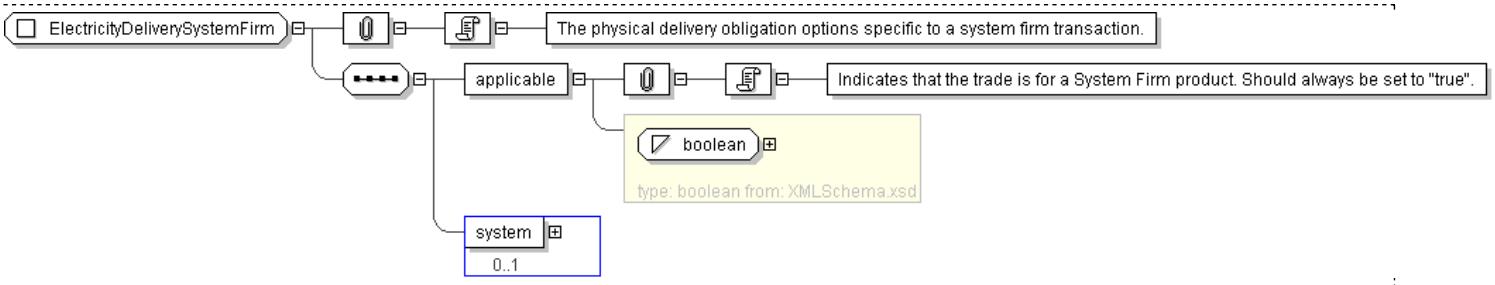
Super-types:	None
Sub-types:	None

Name	ElectricityDeliverySystemFirm
Used by (from the same schema document)	Complex Type ElectricityDelivery
Abstract	no
Documentation	The physical delivery obligation options specific to a system firm transaction.

XML Instance Representation

```
<...>
  <applicable> xsd:boolean </applicable> [1]
   'Indicates that the trade is for a System Firm product. Should always be set to \"true\".'

  <system> CommodityDeliveryPoint </system> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDeliverySystemFirm">
  <xsd:sequence>
    <xsd:element name="applicable" type=" xsd:boolean "/>
    <xsd:element name="system" type=" CommodityDeliveryPoint " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityDeliveryUnitFirm

[Table of contents]

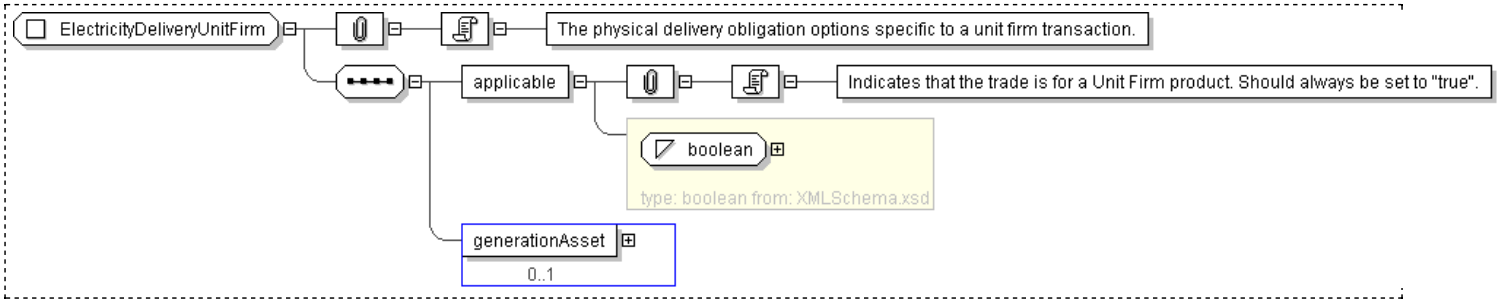
Super-types:	None
Sub-types:	None

Name	ElectricityDeliveryUnitFirm
Used by (from the same schema document)	Complex Type ElectricityDelivery
Abstract	no
Documentation	The physical delivery obligation options specific to a unit firm transaction.

XML Instance Representation

```
<...>
  <applicable> xsd:boolean </applicable> [1]
  'Indicates that the trade is for a Unit Firm product. Should always be set to \"true\".'

  <generationAsset> CommodityDeliveryPoint </generationAsset> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityDeliveryUnitFirm">
  <xsd:sequence>
    <xsd:element name="applicable" type=" xsd:boolean "/>
    <xsd:element name="generationAsset" type=" CommodityDeliveryPoint " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityPhysicalDeliveryQuantity

[Table of contents]

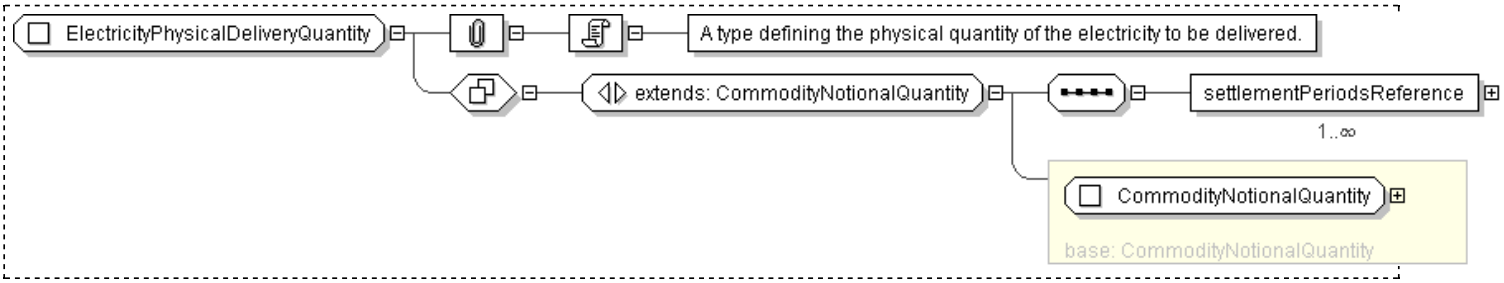
Super-types:	CommodityNotionalQuantity < ElectricityPhysicalDeliveryQuantity (by extension)
Sub-types:	None

Name	ElectricityPhysicalDeliveryQuantity
Used by (from the same schema document)	Complex Type ElectricityPhysicalQuantity
Abstract	no
Documentation	A type defining the physical quantity of the electricity to be delivered.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityUnit> QuantityUnit </quantityUnit> [1]
    'Quantity Unit is the unit of measure applicable for the quantity on the Transaction.'

   <quantityFrequency> CommodityQuantityFrequency </quantityFrequency> [1]
    'The frequency at which the Notional Quantity is deemed to apply for purposes of calculating the
    Total Notional Quantity.'

   <quantity> xsd:decimal </quantity> [1]
    'Amount of commodity per quantity frequency.'

   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
    'A pointer style reference to the range(s) of Settlement Periods to which this quantity applies.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityPhysicalDeliveryQuantity">
  <xsd:complexContent>
    <xsd:extension base=" CommodityNotionalQuantity ">
      <xsd:sequence>
        <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference "
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityPhysicalDeliveryQuantitySchedule

[Table of contents]

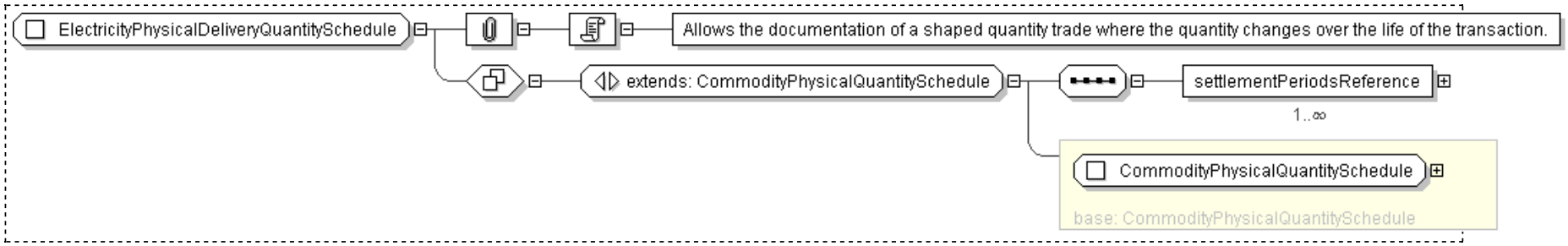
Super-types:	CommodityPhysicalQuantitySchedule < ElectricityPhysicalDeliveryQuantitySchedule (by extension)
Sub-types:	None

Name	ElectricityPhysicalDeliveryQuantitySchedule
Used by (from the same schema document)	Complex Type ElectricityPhysicalQuantity
Abstract	no
Documentation	Allows the documentation of a shaped quantity trade where the quantity changes over the life of the transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityStep> CommodityNotionalQuantity </quantityStep> [1..*]
    'The quantity per Calculation Period. There must be a quantity specified for each Calculation Period, regardless of
    whether the quantity changes or remains the same between periods.'

   Start Choice [1]
      <deliveryPeriodsReference> CalculationPeriodsReference </deliveryPeriodsReference> [1]
      'A pointer style reference to the Delivery Periods defined elsewhere.'

      <deliveryPeriodsScheduleReference> CalculationPeriodsScheduleReference </deliveryPeriodsScheduleReference> [1]
      'A pointer style reference to the Calculation Periods Schedule defined elsewhere.'

   End Choice
   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
    'A pointer style reference to the range(s) of Settlement Periods to which this quantity applies.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityPhysicalDeliveryQuantitySchedule">
  <xsd:complexContent>
    <xsd:extension base=" CommodityPhysicalQuantitySchedule ">
      <xsd:sequence>
        <xsd:element name="                          " type="                          " maxOccurs="          "/>
```

settlementPeriodsReference	**SettlementPeriodsReference**	**unbounded**	

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityPhysicalLeg

[Table of contents]

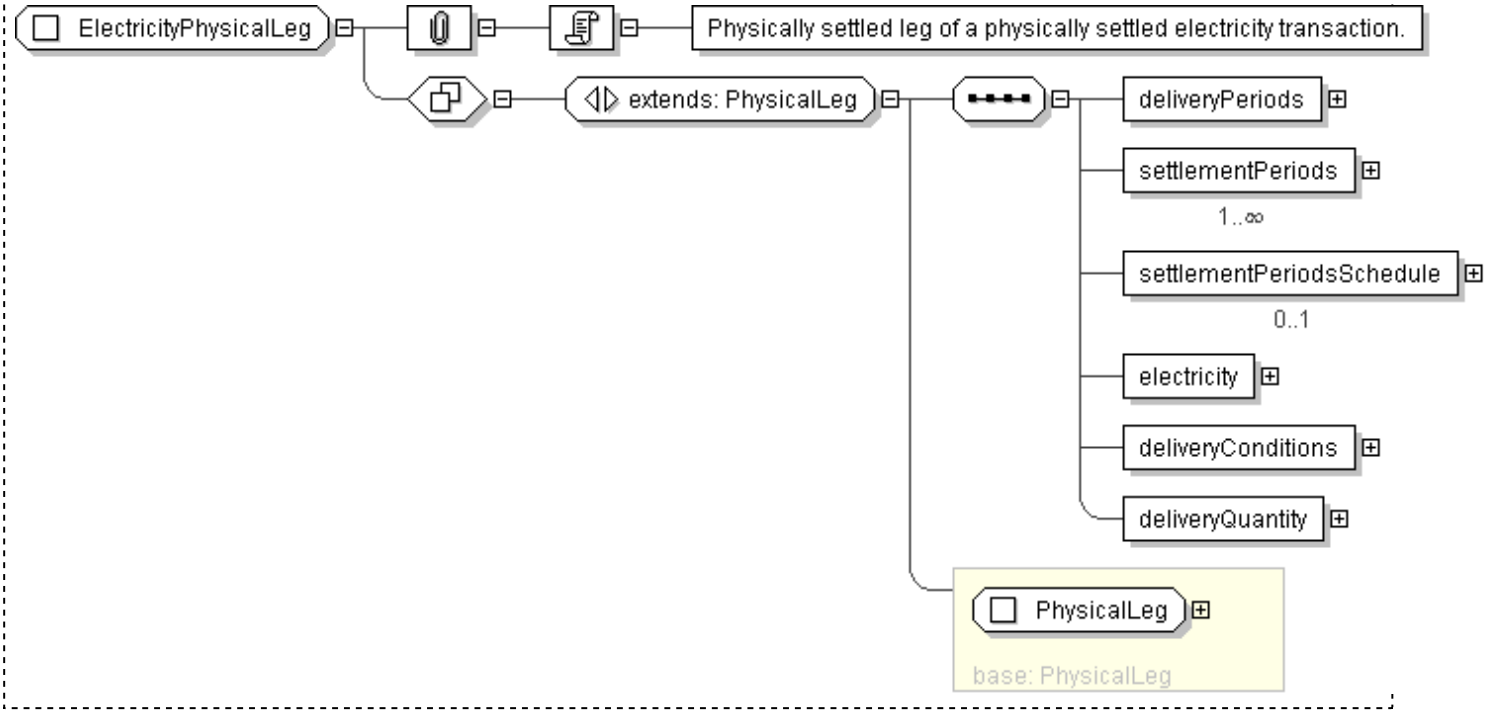
Super-types:	Leg < PhysicalLeg (by extension) < ElectricityPhysicalLeg (by extension)
Sub-types:	None

Name	ElectricityPhysicalLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Physically settled leg of a physically settled electricity transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <deliveryPeriods> CommodityDeliveryPeriods </deliveryPeriods> [1]
   'The different options for specifying the Delivery or Supply Periods. Unless the
   quantity or price is to vary periodically during the trade or physical delivery
   occurs on a periodic basis, periodsSchedule should be used and set to 1T.'

   <settlementPeriods> SettlementPeriods </settlementPeriods> [1..*]
   'The specification of the Settlement Periods in which the electricity will be
   delivered. The Settlement Periods will apply from and including the Effective
   Date up to and including the Termination Date. If more than one
   settlementPeriods element is present this indicates multiple ranges of Settlement
   Periods apply to the entire trade - for example off-peak weekdays and all day
   weekends. Settlement Period ranges should not overlap.'

   <settlementPeriodsSchedule> SettlementPeriodsSchedule </settlementPeriodsSchedule>
   [0..1]
   'The specification of the Settlement Periods in which the electricity will be
   delivered for a \"shaped\" trade i.e. where different Settlement Period ranges
   will apply to different periods of the trade.'

   <electricity> ElectricityProduct </electricity> [1]
   'The specification of the electricity to be delivered.'

   <deliveryConditions> ElectricityDelivery </deliveryConditions> [1]
   'The physical delivery conditions for the transaction.'

   <deliveryQuantity> ElectricityPhysicalQuantity </deliveryQuantity> [1]
   'The different options for specifying the quantity.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityPhysicalLeg">
  <xsd:complexContent>
    <xsd:extension base=" PhysicalLeg ">
      <xsd:sequence>
        <xsd:element name="deliveryPeriods" type=" CommodityDeliveryPeriods "/>
        <xsd:element name="settlementPeriods" type=" SettlementPeriods "
        maxOccurs="unbounded"/>
        <xsd:element name="settlementPeriodsSchedule" type="
        SettlementPeriodsSchedule " minOccurs="0"/>
        <xsd:element name="electricity" type=" ElectricityProduct "/>
        <xsd:element name="deliveryConditions" type=" ElectricityDelivery "/>
        <xsd:element name="deliveryQuantity" type=" ElectricityPhysicalQuantity "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityPhysicalQuantity

[Table of contents]

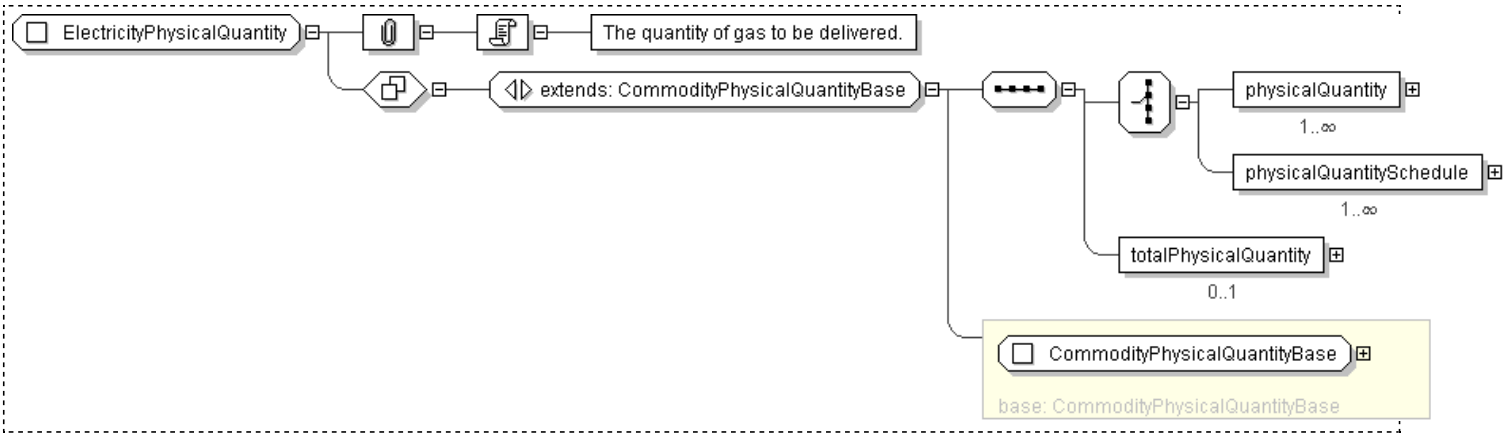
Super-types:	CommodityPhysicalQuantityBase < ElectricityPhysicalQuantity (by extension)
Sub-types:	None

Name	ElectricityPhysicalQuantity
Used by (from the same schema document)	Complex Type ElectricityPhysicalLeg
Abstract	no
Documentation	The quantity of gas to be delivered.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <physicalQuantity> ElectricityPhysicalDeliveryQuantity </physicalQuantity> [1..*]
      'The Quantity per Delivery Period.'

      <physicalQuantitySchedule> ElectricityPhysicalDeliveryQuantitySchedule </physicalQuantitySchedule> [1..*]
      'Allows the documentation of a shaped quantity trade where the quantity changes over the life of the
      transaction. Note that if the range of Settlement Periods also varies over the life of the
      transaction this element should not be used. Instead, physicalQuantity should be repeated for each
      range of Settlement Periods that apply at any point during the trade.'

   End Choice
   <totalPhysicalQuantity> UnitQuantity </totalPhysicalQuantity> [0..1]
   'The Total Quantity of the commodity to be delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityPhysicalQuantity">
  <xsd:complexContent>
    <xsd:extension base=" CommodityPhysicalQuantityBase ">
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="physicalQuantity" type=" ElectricityPhysicalDeliveryQuantity "
          maxOccurs="unbounded"/>
          <xsd:element name="physicalQuantitySchedule" type=" ElectricityPhysicalDeliveryQuantitySchedule "
          maxOccurs="unbounded"/>
        </xsd:choice>
        <xsd:element name="totalPhysicalQuantity" type=" UnitQuantity " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityProduct

[Table of contents]

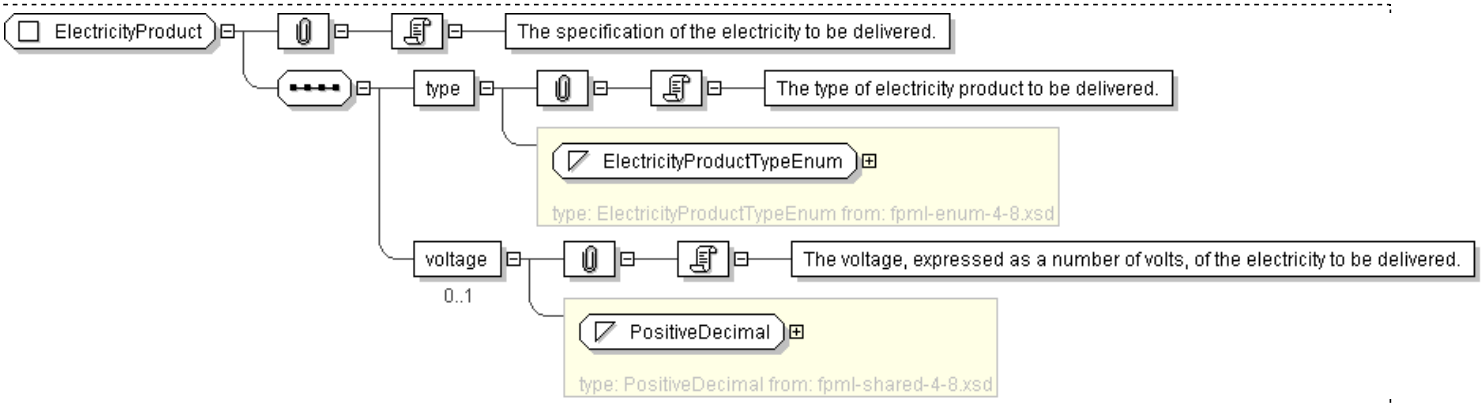
Super-types:	None
Sub-types:	None

Name	ElectricityProduct
Used by (from the same schema document)	Complex Type ElectricityPhysicalLeg
Abstract	no
Documentation	The specification of the electricity to be delivered.

XML Instance Representation

```
<...>
   <type> ElectricityProductTypeEnum </type> [1]
    'The type of electricity product to be delivered.'

   <voltage> PositiveDecimal </voltage> [0..1]
    'The voltage, expressed as a number of volts, of the electricity to be delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityProduct">
   <xsd:sequence>
      <xsd:element name="type" type=" ElectricityProductTypeEnum "/>
      <xsd:element name="voltage" type=" PositiveDecimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityTransmissionContingency

[Table of contents]

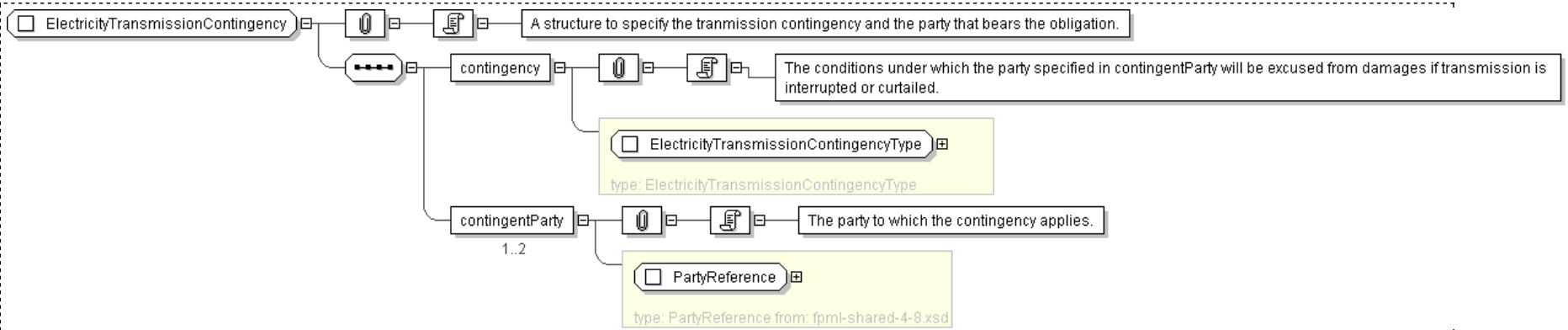
Super-types:	None
Sub-types:	None

Name	ElectricityTransmissionContingency
Used by (from the same schema document)	Complex Type ElectricityDelivery
Abstract	no
Documentation	A structure to specify the tranmission contingency and the party that bears the obligation.

XML Instance Representation

```
<...>
  <contingency> ElectricityTransmissionContingencyType </contingency> [1]
   'The conditions under which the party specified in contingentParty will be excused from damages if transmission is interrupted or curtailed.'

  <contingentParty> PartyReference </contingentParty> [1..2]
   'The party to which the contingency applies.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityTransmissionContingency">
  <xsd:sequence>
    <xsd:element name="contingency" type=" ElectricityTransmissionContingencyType "/>
    <xsd:element name="contingentParty" type=" PartyReference " maxOccurs="2"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ElectricityTransmissionContingencyType

[Table of contents]

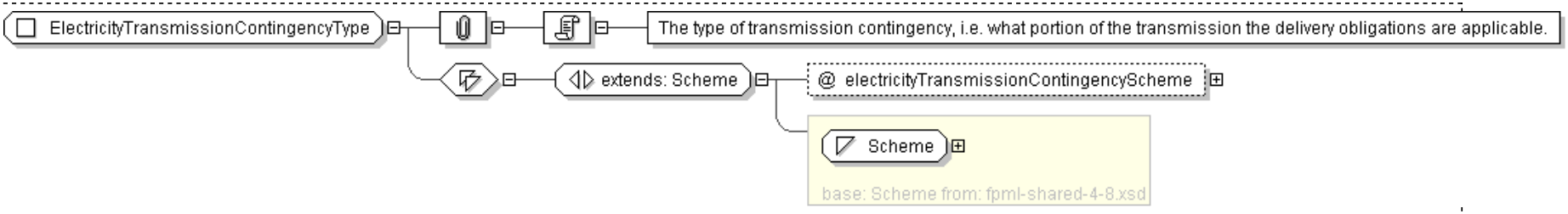
Super-types:	Scheme < ElectricityTransmissionContingencyType (by extension)
Sub-types:	None

Name	ElectricityTransmissionContingencyType
Used by (from the same schema document)	Complex Type ElectricityTransmissionContingency
Abstract	no
Documentation	The type of transmission contingency, i.e. what portion of the transmission the delivery obligations are applicable.

XML Instance Representation

```
<...
 electricityTransmissionContingencyScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ElectricityTransmissionContingencyType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="electricityTransmissionContingencyScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/electricity-transmission-contingency"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedPrice

[Table of contents]

Super-types: None

Sub-types:

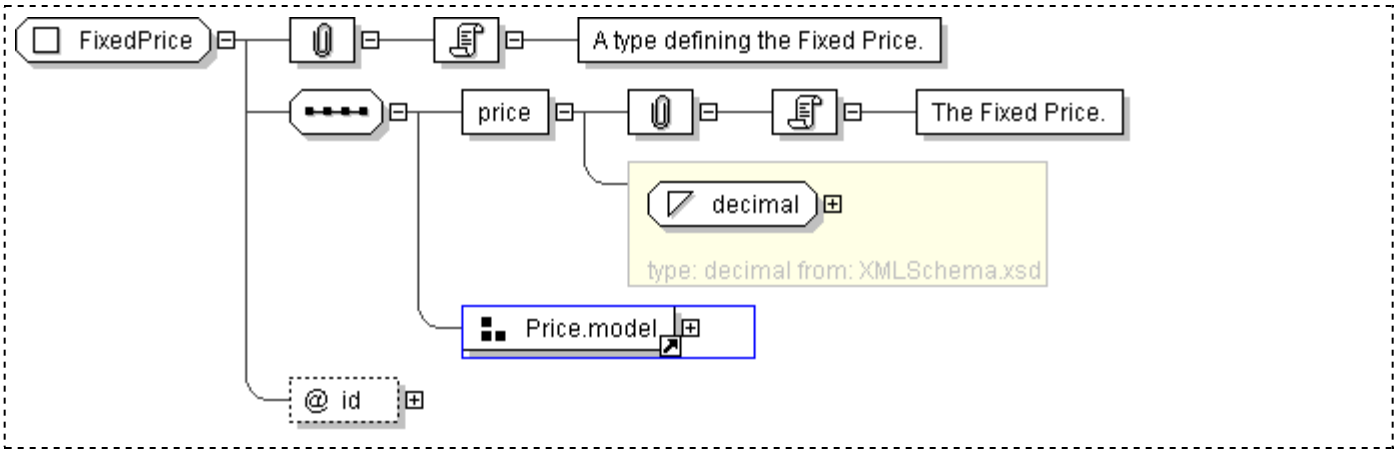
- SettlementPeriodsFixedPrice (by extension)

Name	FixedPrice
Used by (from the same schema document)	Complex Type CommodityFixedPriceSchedule , Complex Type CommoditySettlementPeriodsPriceSchedule , Complex Type NonPeriodicFixedPriceLeg , Model Group CommodityFixedPrice.model
Abstract	no
Documentation	A type defining the Fixed Price.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <price> xsd:decimal </price> [1]
   'The Fixed Price.'

   <priceCurrency> Currency </priceCurrency> [1]
   'Currency of the fixed price.'

   <priceUnit> QuantityUnit </priceUnit> [1]
   'The unit of measure used to calculate the Fixed Price.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedPrice">
   <xsd:sequence>
      <xsd:element name="price" type=" xsd:decimal "/>
      <xsd:group ref=" Price.model "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedPriceLeg

[Table of contents]

Super-types:	Leg < FixedPriceLeg (by extension)
Sub-types:	None

Name	FixedPriceLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Fixed Price Leg of a Commodity Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   Start Choice [1]
      <calculationDates> AdjustableDates </calculationDates> [1]
      'The Calculation Period dates for this leg of the trade where the Calculation
      Periods are all one day long, typically a physically-settled emissions or
      metals trade. Only dates explicitly included determine the Calculation
      Periods and there is a Calculation Period for each date specified.'

      <calculationPeriods> AdjustableDates </calculationPeriods> [1]
      'The Calculation Period start dates for this leg of the swap. This type is
      only intended to be used if the Calculation Periods differ on each leg. If
      Calculation Periods mirror another leg, then the calculationPeriodsReference
      element should be used to point to the Calculation Periods on that leg - or
      the calculationPeriodsScheduleReference can be used to point to the
      Calculation Periods Schedule for that leg.'

      <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule
      </calculationPeriodsSchedule> [1]
      'The Calculation Periods for this leg of the swap. This type is only
      intended to be used if the Calculation Periods differ on each leg. If
      Calculation Periods mirror another leg, then the calculationPeriodsReference
      element should be used to point to the Calculation Periods on the other leg -
      or the calculationPeriodsScheduleReference can be used to point to the
      Calculation Periods Schedule for that leg.'

      Start Choice [1]
         <calculationPeriodsReference> CalculationPeriodsReference
         </calculationPeriodsReference> [1]
         'A pointer style reference to the Calculation Periods defined on another
         leg.'

         <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference
         </calculationPeriodsScheduleReference> [1]
         'A pointer style reference to the Calculation Periods Schedule defined on
         another leg.'
```

<calculationPeriodsDatesReference> CalculationPeriodsDatesReference </calculationPeriodsDatesReference> [1]

'A pointer style reference to single-day-duration Calculation Periods defined on another leg.'

End Choice
End Choice
Start Choice [1]
<fixedPriceSchedule> CommodityFixedPriceSchedule </fixedPriceSchedule> [1]

'Allows the specification of a Fixed Price that varies over the life of the trade.'

Start Choice [1]
<fixedPrice> FixedPrice </fixedPrice> [1]

'Fixed price on which fixed payments are based.'

<worldscaleRate> xsd:decimal </worldscaleRate> [1]

'For a WET Voyager Charter Commodity Swap, the number of Worldscale Points for purposes of the calculation of a Fixed Amount.'

<contractRate> NonNegativeMoney </contractRate> [1]

'For a DRY Voyage Charter or Time Charter Commodity Swap, the price per relevant unit for pruposes of the calculation of a Fixed Amount.'

<settlementPeriodsPrice> SettlementPeriodsFixedPrice </settlementPeriodsPrice> [1..*]

'For an electricity transaction, the fixed price for one or more groups of Settlement Periods on which fixed payments are based. If the fixed price differs for different groups of Settlement Periods, this element should be repeated.'

End Choice
End Choice
<totalPrice> NonNegativeMoney </totalPrice> [0..1]

'The total amount of all fixed payments due during the term of the trade.'

Start Choice [1]
Start Choice [1]
<notionalQuantitySchedule> CommodityNotionalQuantitySchedule </notionalQuantitySchedule> [1]

'Allows the documentation of a shaped notional trade where the notional changes over the life of the transaction.'

<notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]

'The Notional Quantity.'

<settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity </settlementPeriodsNotionalQuantity> [1..*]

'For an electricity transaction, the Notional Quantity for a one or more groups of Settlement Periods to which the Notional Quantity is based. If the schedule differs for different groups of Settlement Periods, this element should be repeated.'

End Choice
<totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]

'The Total Notional Quantity.'

<quantityReference> QuantityReference </quantityReference> [1]

'A pointer style reference to a quantity defined on another leg.'

End Choice
Start Choice [1]
<relativePaymentDates> CommodityRelativePaymentDates </relativePaymentDates> [1]

'The Payment Dates of the trade relative to the Calculation Periods.'

Start Choice [1]
<paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]

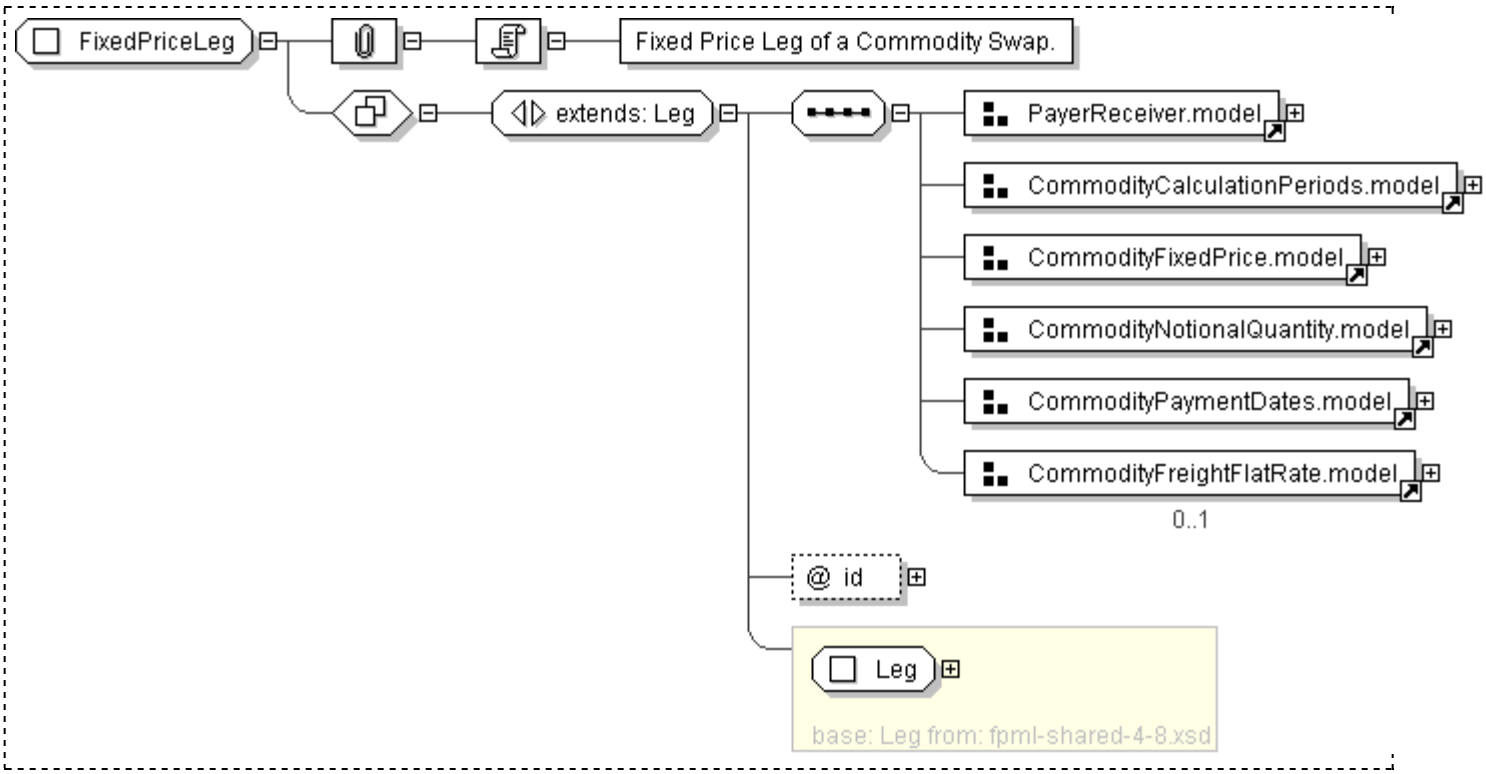
'Dates on which payments will be made.'

<masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]

'If present and true indicates that the Payment Date(s) are specified in

```
        the relevant master agreement.'

      End Choice
   End Choice
   Start Group: CommodityFreightFlatRate.model [0..1]
      <flatRate> FlatRateEnum </flatRate> [1]
      'Whether the Flat Rate is the New Worldwide Tanker Nominal Freight Scale for
      the Freight Index Route taken at the Trade Date of the transaction or taken
      on each Pricing Date.'

      <flatRateAmount> NonNegativeMoney </flatRateAmount> [0..1]
      'If flatRate is set to \"Fixed\", the actual value of the Flat Rate.'

   End Group: CommodityFreightFlatRate.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedPriceLeg">
   <xsd:complexContent>
      <xsd:extension base=" Leg ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:group ref=" CommodityCalculationPeriods.model "/>
            <xsd:group ref=" CommodityFixedPrice.model "/>
            <xsd:group ref=" CommodityNotionalQuantity.model "/>
            <xsd:group ref=" CommodityPaymentDates.model "/>
            <xsd:group ref=" CommodityFreightFlatRate.model " minOccurs="0"/>
         </xsd:sequence>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingLegCalculation

[Table of contents]

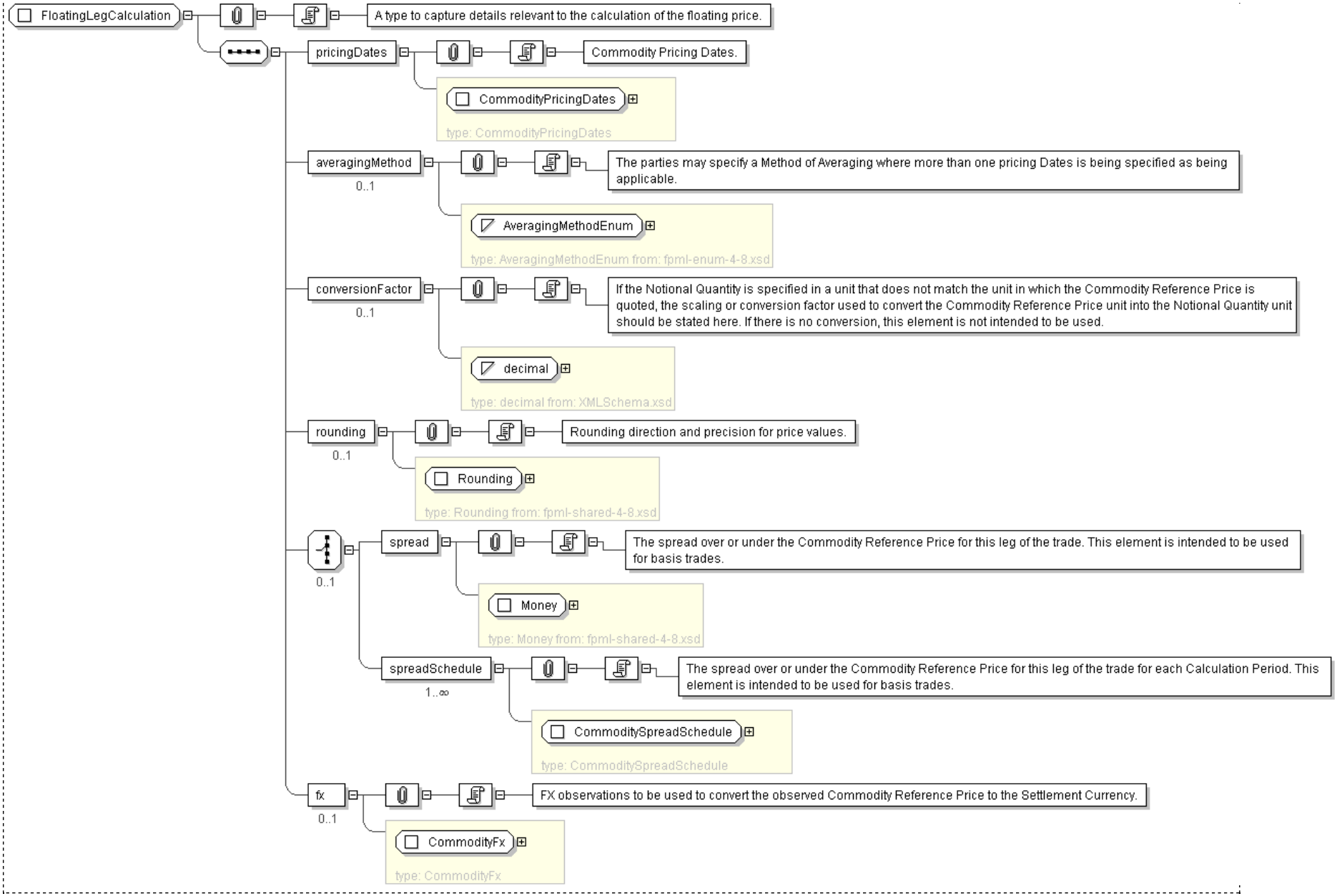
Super-types:	None
Sub-types:	None

Name	FloatingLegCalculation
Used by (from the same schema document)	Complex Type FloatingPriceLeg
Abstract	no
Documentation	A type to capture details relevant to the calculation of the floating price.

XML Instance Representation

```
<...>
   <pricingDates> CommodityPricingDates </pricingDates> [1]
   'Commodity Pricing Dates.'

   <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
   'The parties may specify a Method of Averaging where more than one pricing Dates is being specified as being applicable.'

   <conversionFactor> xsd:decimal </conversionFactor> [0..1]
   'If the Notional Quantity is specified in a unit that does not match the unit in which the Commodity Reference Price is quoted, the scaling or
   conversion factor used to convert the Commodity Reference Price unit into the Notional Quantity unit should be stated here. If there is no
   conversion, this element is not intended to be used.'

   <rounding> Rounding </rounding> [0..1]
   'Rounding direction and precision for price values.'

   Start Choice [0..1]
      <spread> Money </spread> [1]
      'The spread over or under the Commodity Reference Price for this leg of the trade. This element is intended to be used for basis trades.'

      <spreadSchedule> CommoditySpreadSchedule </spreadSchedule> [1..*]
      'The spread over or under the Commodity Reference Price for this leg of the trade for each Calculation Period. This element is intended to be
      used for basis trades.'

   End Choice
   <fx> CommodityFx </fx> [0..1]
   'FX observations to be used to convert the observed Commodity Reference Price to the Settlement Currency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingLegCalculation">
  <xsd:sequence>
    <xsd:element name="pricingDates" type=" CommodityPricingDates "/>
    <xsd:element name="averagingMethod" type=" AveragingMethodEnum " minOccurs="0"/>
    <xsd:element name="conversionFactor" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="rounding" type=" Rounding " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="spread" type=" Money "/>
      <xsd:element name="spreadSchedule" type=" CommoditySpreadSchedule " maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:element name="fx" type=" CommodityFx " minOccurs="0"/>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingPriceLeg

[Table of contents]

Super-types:	Leg < FloatingPriceLeg (by extension)
Sub-types:	None

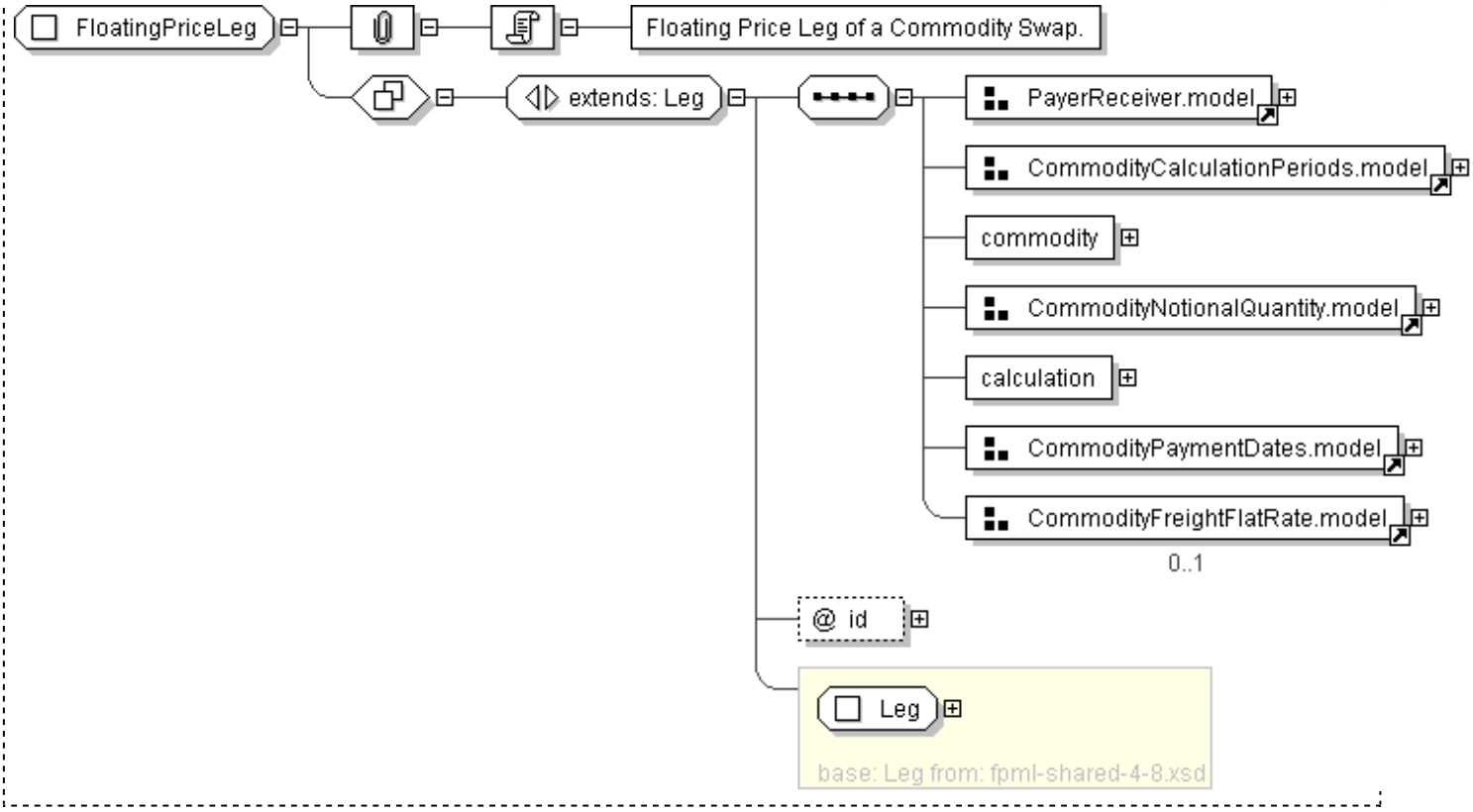
Name	FloatingPriceLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Floating Price Leg of a Commodity Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   Start Choice [1]
      <calculationDates> AdjustableDates </calculationDates> [1]
      'The Calculation Period dates for this leg of the trade where the Calculation
      Periods are all one day long, typically a physically-settled emissions or
      metals trade. Only dates explicitly included determine the Calculation Periods
      and there is a Calculation Period for each date specified.'

      <calculationPeriods> AdjustableDates </calculationPeriods> [1]
      'The Calculation Period start dates for this leg of the swap. This type is
      only intended to be used if the Calculation Periods differ on each leg. If
      Calculation Periods mirror another leg, then the calculationPeriodsReference
      element should be used to point to the Calculation Periods on that leg - or
      the calculationPeriodsScheduleReference can be used to point to the Calculation
      Periods Schedule for that leg.'

      <calculationPeriodsSchedule> CommodityCalculationPeriodsSchedule
      </calculationPeriodsSchedule> [1]
      'The Calculation Periods for this leg of the swap. This type is only intended
      to be used if the Calculation Periods differ on each leg. If Calculation
      Periods mirror another leg, then the calculationPeriodsReference element should
      be used to point to the Calculation Periods on the other leg - or the
      calculationPeriodsScheduleReference can be used to point to the Calculation
      Periods Schedule for that leg.'

      Start Choice [1]
         <calculationPeriodsReference> CalculationPeriodsReference
         </calculationPeriodsReference> [1]
         'A pointer style reference to the Calculation Periods defined on another
         leg.'

         <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference
         </calculationPeriodsScheduleReference> [1]
         'A pointer style reference to the Calculation Periods Schedule defined on
         another leg.'

         <calculationPeriodsDatesReference> CalculationPeriodsDatesReference
```

```
        </calculationPeriodsDatesReference> [1]
        'A pointer style reference to single-day-duration Calculation Periods
        defined on another leg.'

      End Choice
   End Choice
   <commodity> Commodity </commodity> [1]
   'Specifies the underlying instrument. At this time, only underlyers of type
   Commodity are supported; the choice group in the future could offer the
   possibility of adding other types later.'

   Start Choice [1]
      Start Choice [1]
        <notionalQuantitySchedule> CommodityNotionalQuantitySchedule
        </notionalQuantitySchedule> [1]
        'Allows the documentation of a shaped notional trade where the notional
        changes over the life of the transaction.'

        <notionalQuantity> CommodityNotionalQuantity </notionalQuantity> [1]
        'The Notional Quantity.'

        <settlementPeriodsNotionalQuantity> CommoditySettlementPeriodsNotionalQuantity
        </settlementPeriodsNotionalQuantity> [1..*]
        'For an electricity transaction, the Notional Quantity for a one or more
        groups of Settlement Periods to which the Notional Quantity is based. If the
        schedule differs for different groups of Settlement Periods, this element
        should be repeated.'

      End Choice
      <totalNotionalQuantity> xsd:decimal </totalNotionalQuantity> [0..1]
      'The Total Notional Quantity.'

      <quantityReference> QuantityReference </quantityReference> [1]
      'A pointer style reference to a quantity defined on another leg.'

   End Choice
   <calculation> FloatingLegCalculation </calculation> [1]
   'Defines details relevant to the calculation of the floating price.'

   Start Choice [1]
      <relativePaymentDates> CommodityRelativePaymentDates </relativePaymentDates> [1]
      'The Payment Dates of the trade relative to the Calculation Periods.'

      Start Choice [1]
        <paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]
        'Dates on which payments will be made.'

        <masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]
        'If present and true indicates that the Payment Date(s) are specified in the
        relevant master agreement.'

      End Choice
   End Choice
   Start Group: CommodityFreightFlatRate.model [0..1]
      <flatRate> FlatRateEnum </flatRate> [1]
      'Whether the Flat Rate is the New Worldwide Tanker Nominal Freight Scale for
      the Freight Index Route taken at the Trade Date of the transaction or taken on
      each Pricing Date.'

      <flatRateAmount> NonNegativeMoney </flatRateAmount> [0..1]
      'If flatRate is set to \"Fixed\", the actual value of the Flat Rate.'

   End Group: CommodityFreightFlatRate.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingPriceLeg">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:group ref=" CommodityCalculationPeriods.model "/>
        <xsd:element name="commodity" type=" Commodity "/>
        <xsd:group ref=" CommodityNotionalQuantity.model "/>
        <xsd:element name="calculation" type=" FloatingLegCalculation "/>
        <xsd:group ref=" CommodityPaymentDates.model "/>
        <xsd:group ref=" CommodityFreightFlatRate.model " minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasDelivery

[Table of contents]

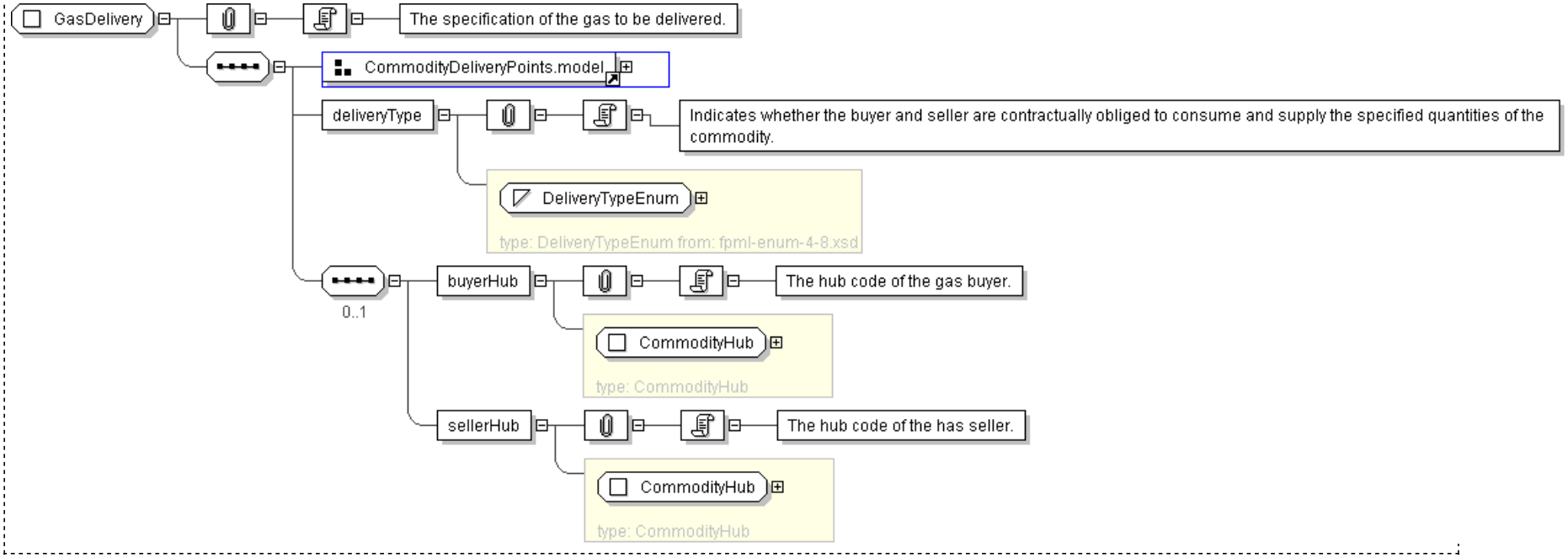
Super-types:	None
Sub-types:	None

Name	GasDelivery
Used by (from the same schema document)	Complex Type GasPhysicalLeg
Abstract	no
Documentation	The specification of the gas to be delivered.

XML Instance Representation

```
<...>
   Start Choice [1]
      <deliveryPoint> GasDeliveryPoint </deliveryPoint> [1]
      'The physical or virtual point at which the commodity will be delivered.'

      <entryPoint> CommodityDeliveryPoint </entryPoint> [1]
      'The physical or virtual point at which the commodity enters a transportation system.'

      <withdrawalPoint> CommodityDeliveryPoint </withdrawalPoint> [1]
      'The physical or virtual point at which the commodity is withdrawn from a transportation system.'

   End Choice
   <deliveryType> DeliveryTypeEnum </deliveryType> [1]
   'Indicates whether the buyer and seller are contractually obliged to consume and supply the specified quantities of the
   commodity.'

   Start Sequence [0..1]
      <buyerHub> CommodityHub </buyerHub> [1]
      'The hub code of the gas buyer.'

      <sellerHub> CommodityHub </sellerHub> [1]
      'The hub code of the has seller.'

   End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasDelivery">
  <xsd:sequence>
    <xsd:group ref=" CommodityDeliveryPoints.model "/>
    <xsd:element name="deliveryType" type=" DeliveryTypeEnum "/>
    <xsd:sequence minOccurs="0">
      <xsd:element name="buyerHub" type=" CommodityHub "/>
      <xsd:element name="sellerHub" type=" CommodityHub "/>
    </xsd:sequence>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasDeliveryPeriods

[Table of contents]

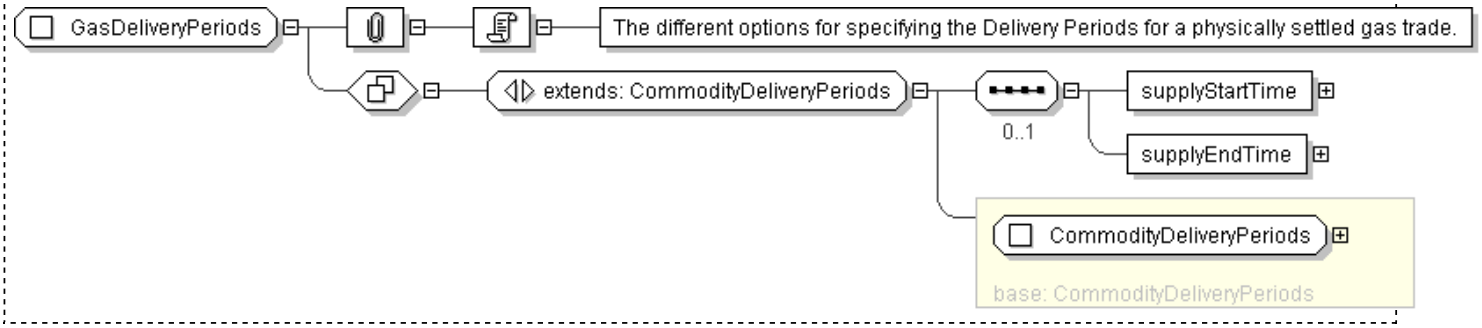
Super-types:	CommodityDeliveryPeriods < GasDeliveryPeriods (by extension)
Sub-types:	None

Name	GasDeliveryPeriods
Used by (from the same schema document)	Complex Type GasPhysicalLeg
Abstract	no
Documentation	The different options for specifying the Delivery Periods for a physically settled gas trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <periods> AdjustableDates </periods> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if
      the Delivery Periods differ from the Calculation Periods on the fixed or floating leg. If
      DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be
      used to point to the Calculation Periods on that leg - or the
      calculationPeriodsScheduleReference can be used to point to the Calculation Periods
      Schedule for that leg.'

      <periodsSchedule> CommodityCalculationPeriodsSchedule </periodsSchedule> [1]
      'The Delivery Periods for this leg of the swap. This type is only intended to be used if
      the Delivery Periods differ from the Calculation Periods on the fixed or floating leg. If
      DeliveryPeriods mirror another leg, then the calculationPeriodsReference element should be
      used to point to the Calculation Periods on that leg - or the
      calculationPeriodsScheduleReference can be used to point to the Calculation Periods
      Schedule for that leg.'

      Start Choice [1]
         <calculationPeriodsReference> CalculationPeriodsReference </calculationPeriodsReference> [1]
         'A pointer style reference to the Calculation Periods defined on another leg.'

         <calculationPeriodsScheduleReference> CalculationPeriodsScheduleReference
         </calculationPeriodsScheduleReference> [1]
         'A pointer style reference to the Calculation Periods Schedule defined on another leg.'

         <calculationPeriodsDatesReference> CalculationPeriodsDatesReference
         </calculationPeriodsDatesReference> [1]
         'A pointer style reference to single-day-duration Calculation Periods defined on
         another leg.'

      End Choice
   End Choice
   Start Sequence [0..1]
      <supplyStartTime> PrevailingTime </supplyStartTime> [1]
      'The time at which gas delivery should start on each day of the Delivery Period(s).'

      <supplyEndTime> PrevailingTime </supplyEndTime> [1]
      'The time at which gas delivery should end on each day of the Delivery Period(s).'

   End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasDeliveryPeriods">
  <xsd:complexContent>
    <xsd:extension base=" CommodityDeliveryPeriods ">
      <xsd:sequence minOccurs="0">
        <xsd:element name="supplyStartTime" type=" PrevailingTime "/>
        <xsd:element name="supplyEndTime" type=" PrevailingTime "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasDeliveryPoint

[Table of contents]

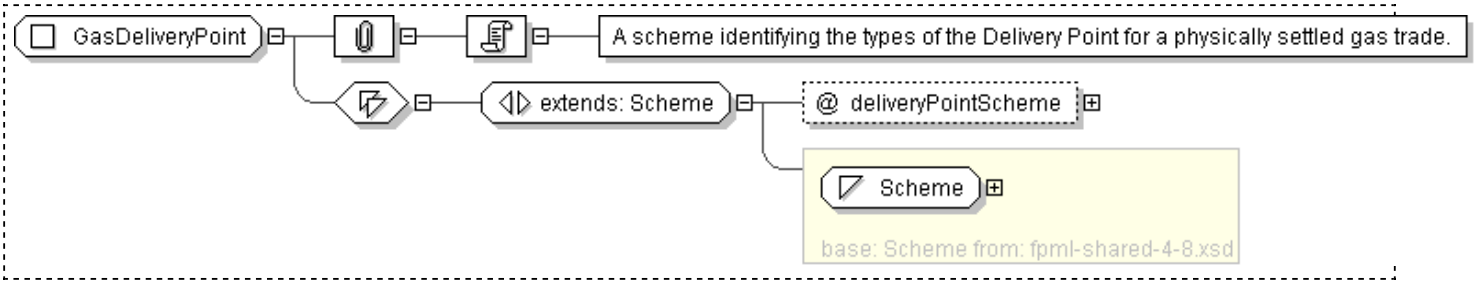
Super-types:	Scheme < GasDeliveryPoint (by extension)
Sub-types:	None

Name	GasDeliveryPoint
Used by (from the same schema document)	Model Group CommodityDeliveryPoints.model
Abstract	no
Documentation	A scheme identifying the types of the Delivery Point for a physically settled gas trade.

XML Instance Representation

```
<...
 deliveryPointScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasDeliveryPoint">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="deliveryPointScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasPhysicalLeg

[Table of contents]

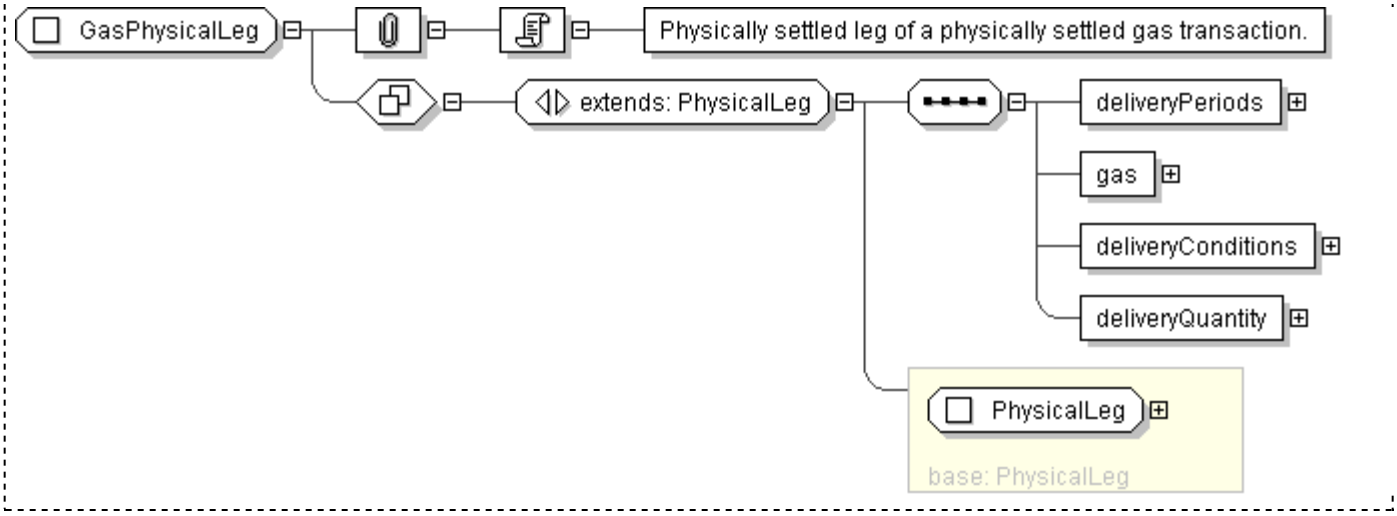
Super-types:	Leg < PhysicalLeg (by extension) < GasPhysicalLeg (by extension)
Sub-types:	None

Name	GasPhysicalLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Physically settled leg of a physically settled gas transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <deliveryPeriods> GasDeliveryPeriods </deliveryPeriods> [1]
   'The different options for specifying the Delivery or Supply Periods. Unless
   the quantity or price is to vary periodically during the trade or physical
   delivery occurs on a periodic basis, periodsSchedule should be used and set to
   1T.'

   <gas> GasProduct </gas> [1]
   'The specification of the gas to be delivered.'

   <deliveryConditions> GasDelivery </deliveryConditions> [1]
   'The physical delivery conditions for the transaction.'

   <deliveryQuantity> GasPhysicalQuantity </deliveryQuantity> [1]
   'The different options for specifying the quantity. For Fixed trades where the
   quantity is known at the time of confirmation, a single quantity or a quantity
   per Delivery Period may be specified. For Variable trades minimum and maximum
   trades may be specified.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasPhysicalLeg">
  <xsd:complexContent>
    <xsd:extension base=" PhysicalLeg ">
      <xsd:sequence>
        <xsd:element name="deliveryPeriods" type=" GasDeliveryPeriods "/>
        <xsd:element name="gas" type=" GasProduct "/>
        <xsd:element name="deliveryConditions" type=" GasDelivery "/>
        <xsd:element name="deliveryQuantity" type=" GasPhysicalQuantity "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasPhysicalQuantity

[Table of contents]

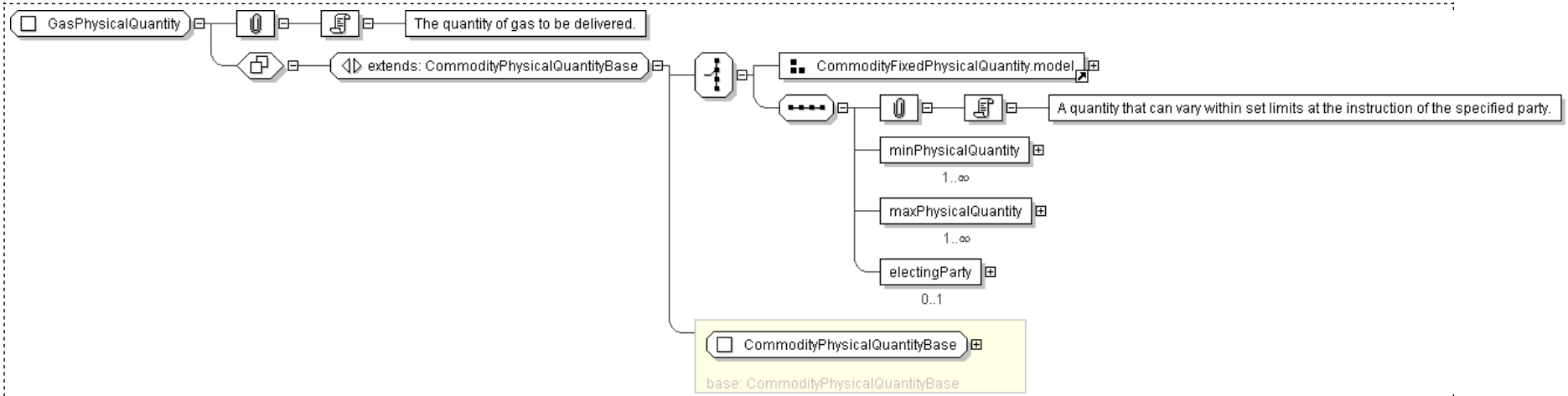
Super-types:	CommodityPhysicalQuantityBase < GasPhysicalQuantity (by extension)
Sub-types:	None

Name	GasPhysicalQuantity
Used by (from the same schema document)	Complex Type GasPhysicalLeg
Abstract	no
Documentation	The quantity of gas to be delivered.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      Start Choice [1]
         <physicalQuantity> CommodityNotionalQuantity </physicalQuantity> [1]
          'The Quantity per Delivery Period.'

         <physicalQuantitySchedule> CommodityPhysicalQuantitySchedule </physicalQuantitySchedule> [1]
          'Allows the documentation of a shaped quantity trade where the quantity changes over the life of the transaction.'

      End Choice
      <totalPhysicalQuantity> UnitQuantity </totalPhysicalQuantity> [0..1]
       'The Total Quantity of the commodity to be delivered.'

      <minPhysicalQuantity> CommodityNotionalQuantity </minPhysicalQuantity> [1..*]
       'The minimum quantity to be delivered. If separate minimums need to be specified for different periods (e.g. a minimum per day and a minimum per
       month) this element should be repeated.'

      <maxPhysicalQuantity> CommodityNotionalQuantity </maxPhysicalQuantity> [1..*]
       'The maximum quantity to be delivered. If separate minimums need to be specified for different periods (e.g. a minimum per day and a minimum per
       month) this element should be repeated.'

      <electingParty> PartyReference </electingParty> [0..1]
       'Indicates the party able to choose whether the gas is delivered for a particular period e.g. a swing or interruptible contract.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasPhysicalQuantity">
   <xsd:complexContent>
      <xsd:extension base=" CommodityPhysicalQuantityBase ">
         <xsd:choice>
            <xsd:group ref=" CommodityFixedPhysicalQuantity.model "/>
            <xsd:sequence>
               <xsd:element name="minPhysicalQuantity" type=" CommodityNotionalQuantity " maxOccurs="unbounded"/>
               <xsd:element name="maxPhysicalQuantity" type=" CommodityNotionalQuantity " maxOccurs="unbounded"/>
               <xsd:element name="electingParty" type=" PartyReference " minOccurs="0"/>
            </xsd:sequence>
         </xsd:choice>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasProduct

[Table of contents]

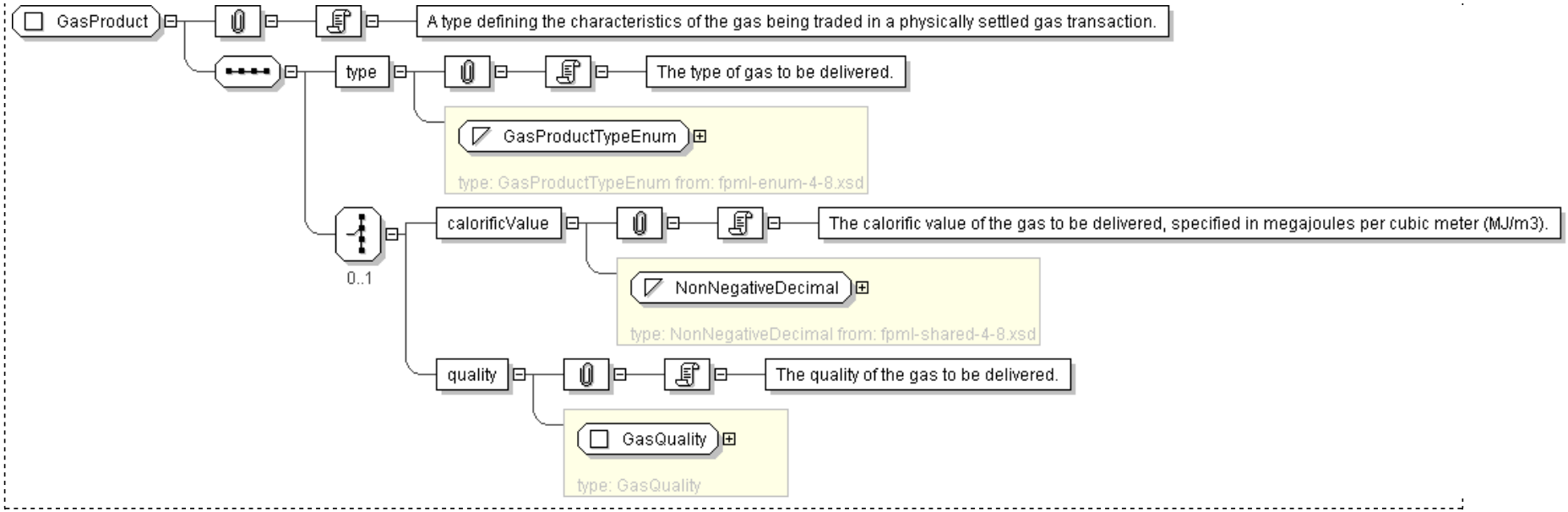
Super-types:	None
Sub-types:	None

Name	GasProduct
Used by (from the same schema document)	Complex Type GasPhysicalLeg
Abstract	no
Documentation	A type defining the characteristics of the gas being traded in a physically settled gas transaction.

XML Instance Representation

```
<...>
   <type> GasProductTypeEnum </type> [1]
   'The type of gas to be delivered.'

   Start Choice [0..1]
      <calorificValue> NonNegativeDecimal </calorificValue> [1]
      'The calorific value of the gas to be delivered, specified in megajoules per cubic meter (MJ/m3).'

      <quality> GasQuality </quality> [1]
      'The quality of the gas to be delivered.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasProduct">
  <xsd:sequence>
    <xsd:element name="type" type=" GasProductTypeEnum "/>
    <xsd:choice minOccurs="0">
      <xsd:element name="calorificValue" type=" NonNegativeDecimal "/>
      <xsd:element name="quality" type=" GasQuality "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GasQuality

[Table of contents]

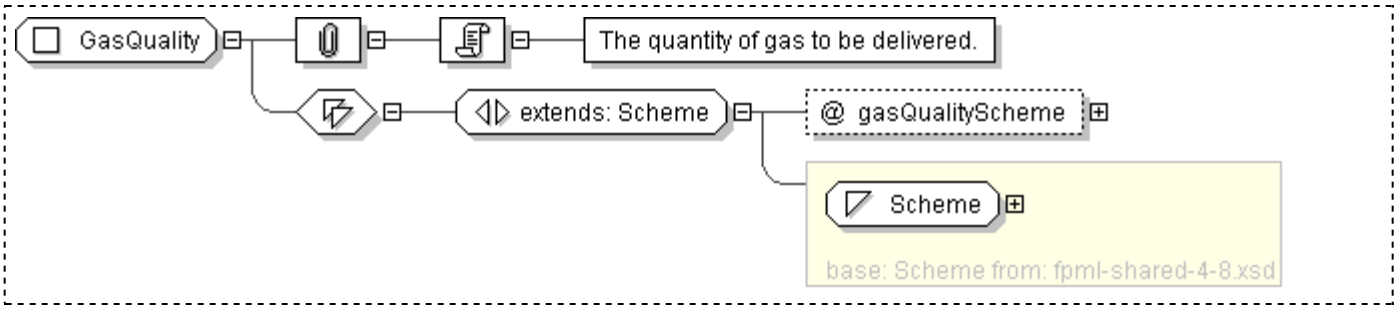
Super-types:	Scheme < GasQuality (by extension)
Sub-types:	None

Name	GasQuality
Used by (from the same schema document)	Complex Type GasProduct
Abstract	no
Documentation	The quantity of gas to be delivered.

XML Instance Representation

```
<...
 gasQualityScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GasQuality">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="gasQualityScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-gas-quality"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Lag

[Table of contents]

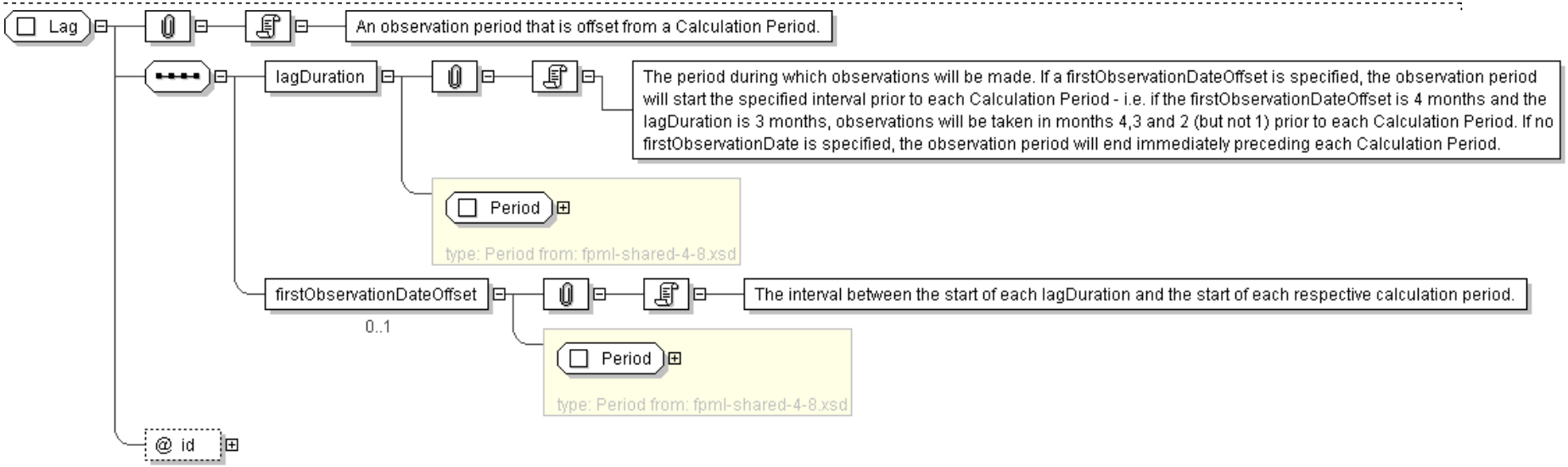
Super-types:	None
Sub-types:	None

Name	Lag
Used by (from the same schema document)	Complex Type CommodityPricingDates , Model Group LagOrReference.model
Abstract	no
Documentation	An observation period that is offset from a Calculation Period.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <lagDuration> Period </lagDuration> [1]
    'The period during which observations will be made. If a firstObservationDateOffset is specified, the observation period will
    start the specified interval prior to each Calculation Period - i.e. if the firstObservationDateOffset is 4 months and the
    lagDuration is 3 months, observations will be taken in months 4,3 and 2 (but not 1) prior to each Calculation Period. If no
    firstObservationDate is specified, the observation period will end immediately preceding each Calculation Period.'

   <firstObservationDateOffset> Period </firstObservationDateOffset> [0..1]
    'The interval between the start of each lagDuration and the start of each respective calculation period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Lag">
   <xsd:sequence>
      <xsd:element name="lagDuration" type=" Period "/>
      <xsd:element name="firstObservationDateOffset" type=" Period " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LagReference

[Table of contents]

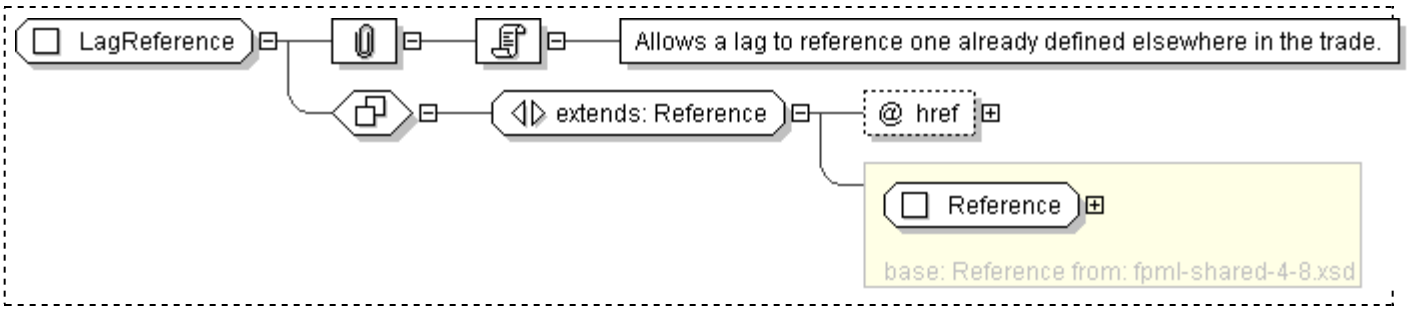
Super-types:	Reference < LagReference (by extension)
Sub-types:	None

Name	LagReference
Used by (from the same schema document)	Model Group LagOrReference.model
Abstract	no
Documentation	Allows a lag to reference one already defined elsewhere in the trade.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LagReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="Lag"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MarketDisruptionEvent

[Table of contents]

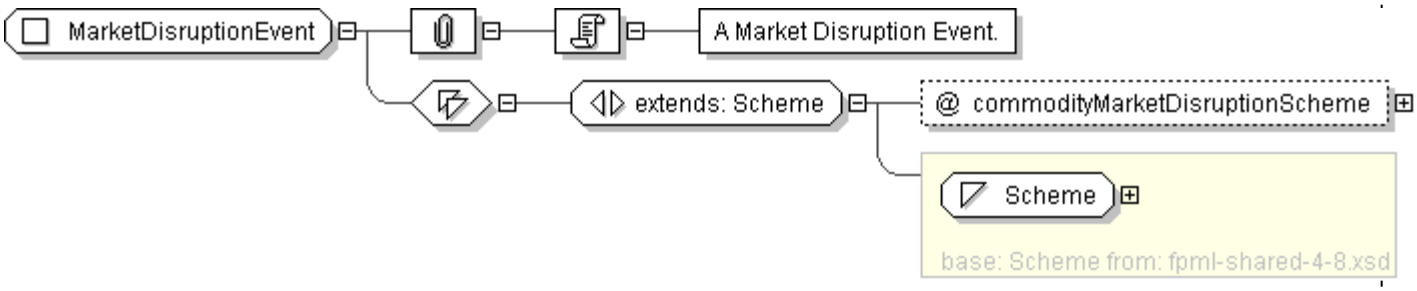
Super-types:	Scheme < MarketDisruptionEvent (by extension)
Sub-types:	None

Name	MarketDisruptionEvent
Used by (from the same schema document)	Complex Type CommodityMarketDisruption , Complex Type CommodityMarketDisruption
Abstract	no
Documentation	A Market Disruption Event.

XML Instance Representation

```
<...
 commodityMarketDisruptionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MarketDisruptionEvent">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityMarketDisruptionScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-market-disruption"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonPeriodicFixedPriceLeg

[Table of contents]

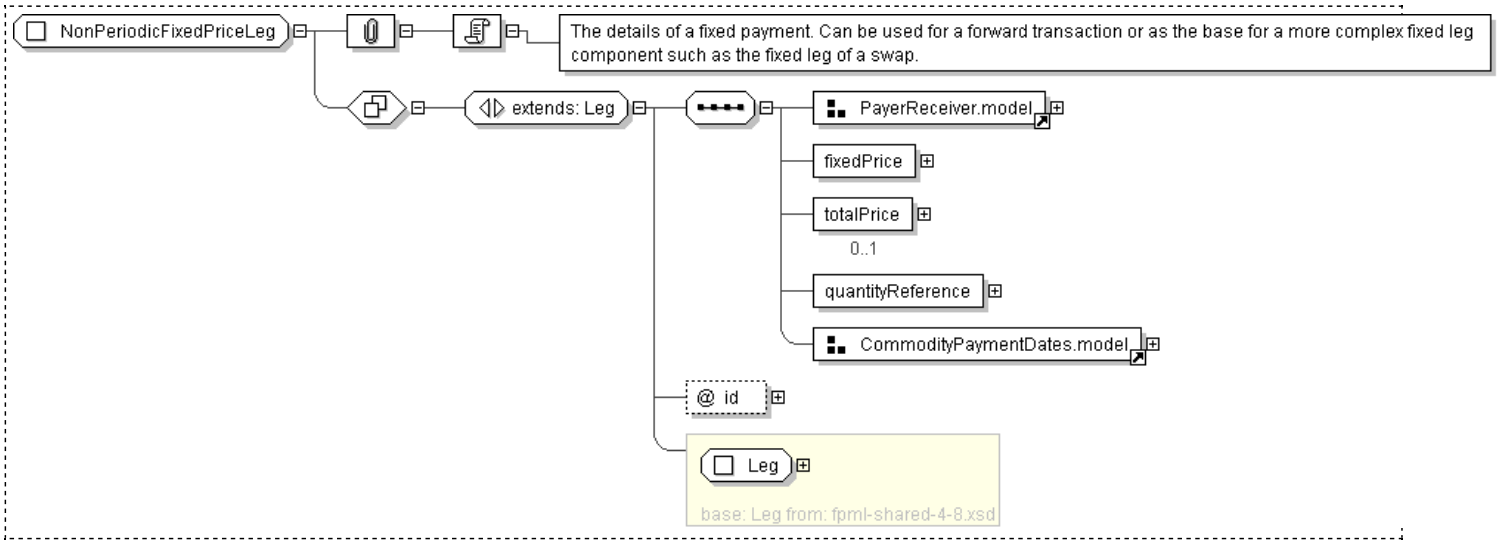
Super-types:	Leg < NonPeriodicFixedPriceLeg (by extension)
Sub-types:	None

Name	NonPeriodicFixedPriceLeg
Used by (from the same schema document)	Complex Type CommodityForward
Abstract	no
Documentation	The details of a fixed payment. Can be used for a forward transaction or as the base for a more complex fixed leg component such as the fixed leg of a swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <fixedPrice> FixedPrice </fixedPrice> [1]
   'Fixed price on which fixed payments are based.'

   <totalPrice> NonNegativeMoney </totalPrice> [0..1]
   'The total amount of the fixed payment for all units of the underlying commodity.'

   <quantityReference> QuantityReference </quantityReference> [1]
   'A pointer style reference to a quantity defined on another leg.'

   Start Choice [1]
      <relativePaymentDates> CommodityRelativePaymentDates </relativePaymentDates> [1]
      'The Payment Dates of the trade relative to the Calculation Periods.'

      Start Choice [1]
         <paymentDates> AdjustableDatesOrRelativeDateOffset </paymentDates> [1]
         'Dates on which payments will be made.'

         <masterAgreementPaymentDates> xsd:boolean </masterAgreementPaymentDates> [1]
         'If present and true indicates that the Payment Date(s) are specified in the relevant master agreement.'

      End Choice
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonPeriodicFixedPriceLeg">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
```

```
          <xsd:element name="fixedPrice" type=" FixedPrice "/>
          <xsd:element name="totalPrice" type=" NonNegativeMoney " minOccurs="0"/>
          <xsd:element name="quantityReference" type=" QuantityReference "/>
          <xsd:group ref=" CommodityPaymentDates.model "/>
        </xsd:sequence>
        <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilDelivery

[Table of contents]

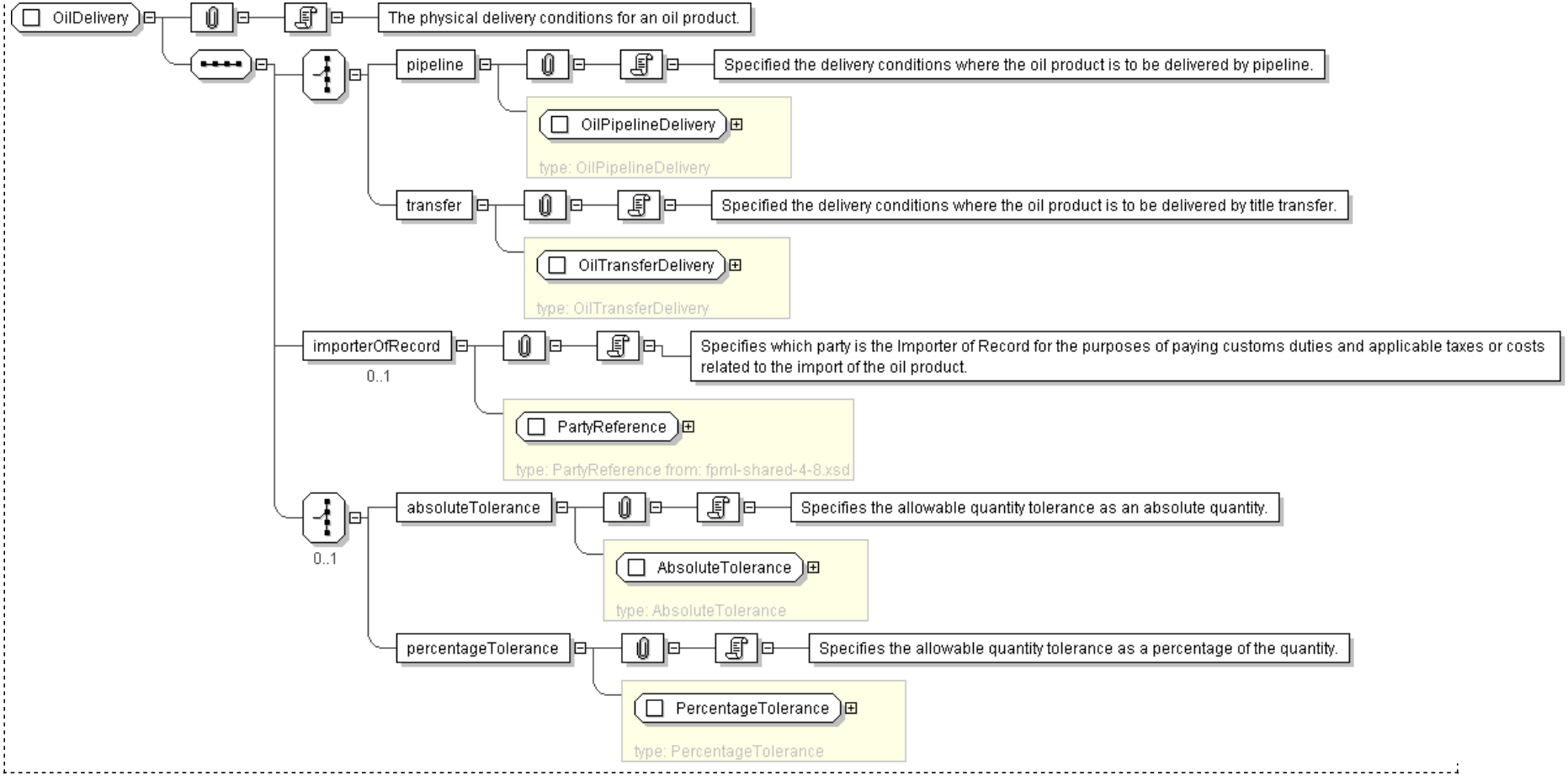
Super-types:	None
Sub-types:	None

Name	OilDelivery
Used by (from the same schema document)	Complex Type OilPhysicalLeg
Abstract	no
Documentation	The physical delivery conditions for an oil product.

XML Instance Representation

```
<...>
   Start Choice [1]
      <pipeline> OilPipelineDelivery </pipeline> [1]
      'Specified the delivery conditions where the oil product is to be delivered by pipeline.'

      <transfer> OilTransferDelivery </transfer> [1]
      'Specified the delivery conditions where the oil product is to be delivered by title transfer.'

   End Choice
   <importerOfRecord> PartyReference </importerOfRecord> [0..1]
   'Specifies which party is the Importer of Record for the purposes of paying customs duties and applicable taxes or costs related to
   the import of the oil product.'

   Start Choice [0..1]
      <absoluteTolerance> AbsoluteTolerance </absoluteTolerance> [1]
      'Specifies the allowable quantity tolerance as an absolute quantity.'

      <percentageTolerance> PercentageTolerance </percentageTolerance> [1]
      'Specifies the allowable quantity tolerance as a percentage of the quantity.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilDelivery">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="pipeline" type=" OilPipelineDelivery "/>
      <xsd:element name="transfer" type=" OilTransferDelivery "/>
    </xsd:choice>
    <xsd:element name="importerOfRecord" type=" PartyReference " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="absoluteTolerance" type=" AbsoluteTolerance "/>
      <xsd:element name="percentageTolerance" type=" PercentageTolerance "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilPhysicalLeg

[Table of contents]

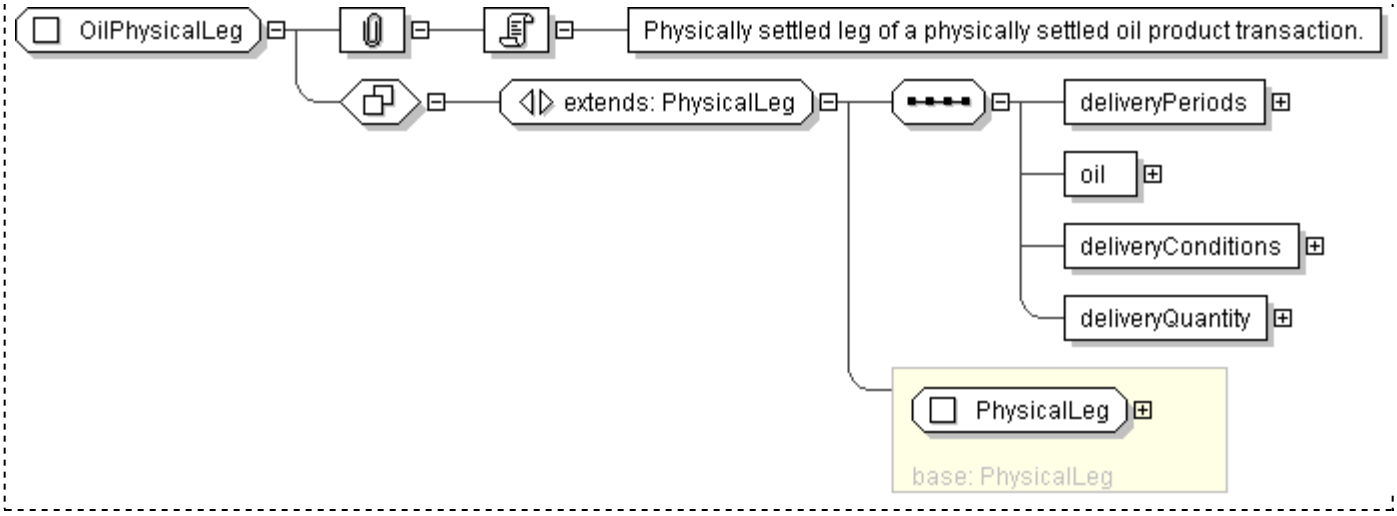
Super-types:	Leg < PhysicalLeg (by extension) < OilPhysicalLeg (by extension)
Sub-types:	None

Name	OilPhysicalLeg
Used by (from the same schema document)	Complex Type CommoditySwap
Abstract	no
Documentation	Physically settled leg of a physically settled oil product transaction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <deliveryPeriods> CommodityDeliveryPeriods </deliveryPeriods> [1]
   'The different options for specifying the Delivery or Supply Periods. Unless
   the quantity or price is to vary periodically during the trade or physical
   delivery occurs on a periodic basis, periodsSchedule should be used and set to
   1T.'

   <oil> OilProduct </oil> [1]
   'The specification of the oil product to be delivered.'

   <deliveryConditions> OilDelivery </deliveryConditions> [1]
   'The physical delivery conditions for the transaction.'

   <deliveryQuantity> CommodityPhysicalQuantity </deliveryQuantity> [1]
   'The different options for specifying the quantity.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilPhysicalLeg">
  <xsd:complexContent>
    <xsd:extension base=" PhysicalLeg ">
      <xsd:sequence>
        <xsd:element name="deliveryPeriods" type=" CommodityDeliveryPeriods "/>
        <xsd:element name="oil" type=" OilProduct "/>
        <xsd:element name="deliveryConditions" type=" OilDelivery "/>
        <xsd:element name="deliveryQuantity" type=" CommodityPhysicalQuantity "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilPipelineDelivery

[Table of contents]

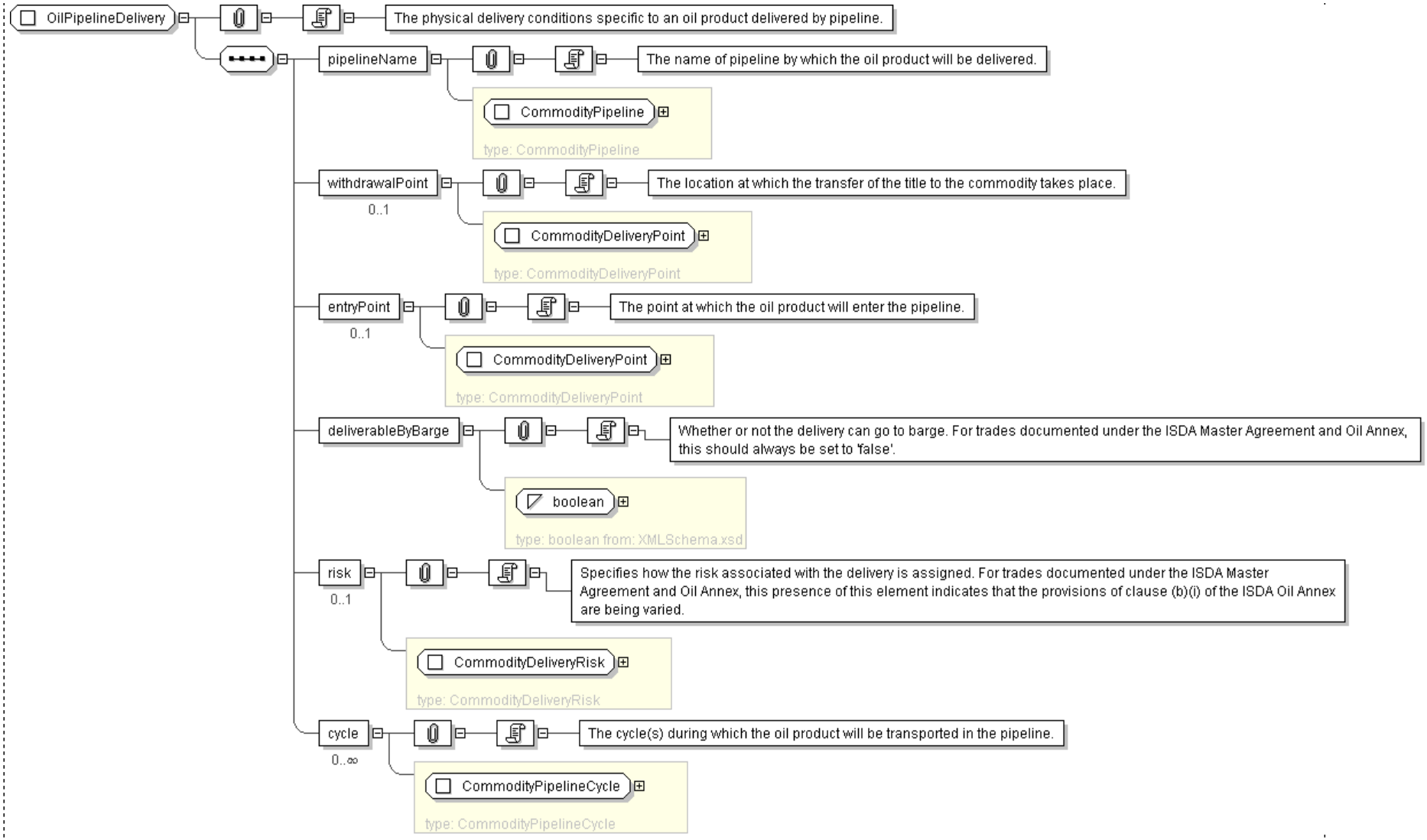
Super-types:	None
Sub-types:	None

Name	OilPipelineDelivery
Used by (from the same schema document)	Complex Type OilDelivery
Abstract	no
Documentation	The physical delivery conditions specific to an oil product delivered by pipeline.

XML Instance Representation

```
<...>
   <pipelineName> CommodityPipeline </pipelineName> [1]
   'The name of pipeline by which the oil product will be delivered.'

   <withdrawalPoint> CommodityDeliveryPoint </withdrawalPoint> [0..1]
   'The location at which the transfer of the title to the commodity takes place.'

   <entryPoint> CommodityDeliveryPoint </entryPoint> [0..1]
   'The point at which the oil product will enter the pipeline.'

   <deliverableByBarge> xsd:boolean </deliverableByBarge> [1]
   'Whether or not the delivery can go to barge. For trades documented under the ISDA Master Agreement and Oil Annex, this should always be
   set to \'false\'.'

   <risk> CommodityDeliveryRisk </risk> [0..1]
   'Specifies how the risk associated with the delivery is assigned. For trades documented under the ISDA Master Agreement and Oil Annex,
   this presence of this element indicates that the provisions of clause (b)(i) of the ISDA Oil Annex are being varied.'

   <cycle> CommodityPipelineCycle </cycle> [0..*]
   'The cycle(s) during which the oil product will be transported in the pipeline.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilPipelineDelivery">
  <xsd:sequence>
    <xsd:element name="pipelineName" type=" CommodityPipeline "/>
    <xsd:element name="withdrawalPoint" type=" CommodityDeliveryPoint " minOccurs="0"/>
    <xsd:element name="entryPoint" type=" CommodityDeliveryPoint " minOccurs="0"/>
    <xsd:element name="deliverableByBarge" type=" xsd:boolean "/>
    <xsd:element name="risk" type=" CommodityDeliveryRisk " minOccurs="0"/>
    <xsd:element name="cycle" type=" CommodityPipelineCycle " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilProduct

[Table of contents]

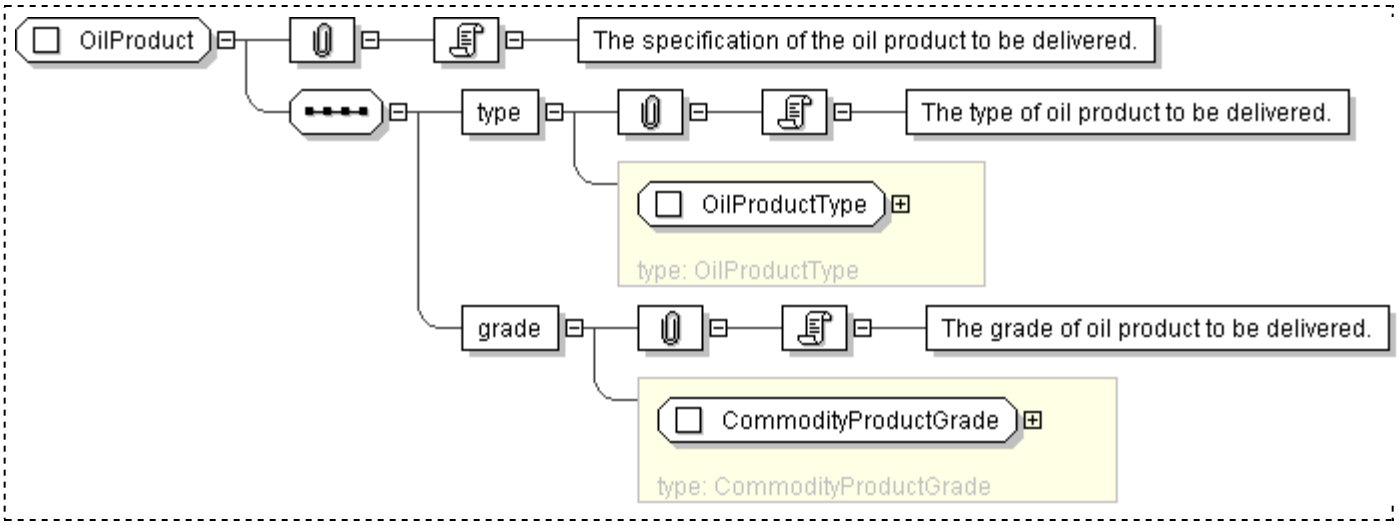
Super-types:	None
Sub-types:	None

Name	OilProduct
Used by (from the same schema document)	Complex Type OilPhysicalLeg
Abstract	no
Documentation	The specification of the oil product to be delivered.

XML Instance Representation

```
<...>
   <type> OilProductType </type> [1]
   'The type of oil product to be delivered.'

   <grade> CommodityProductGrade </grade> [1]
   'The grade of oil product to be delivered.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilProduct">
   <xsd:sequence>
      <xsd:element name="type" type=" OilProductType "/>
      <xsd:element name="grade" type=" CommodityProductGrade "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilProductType

[Table of contents]

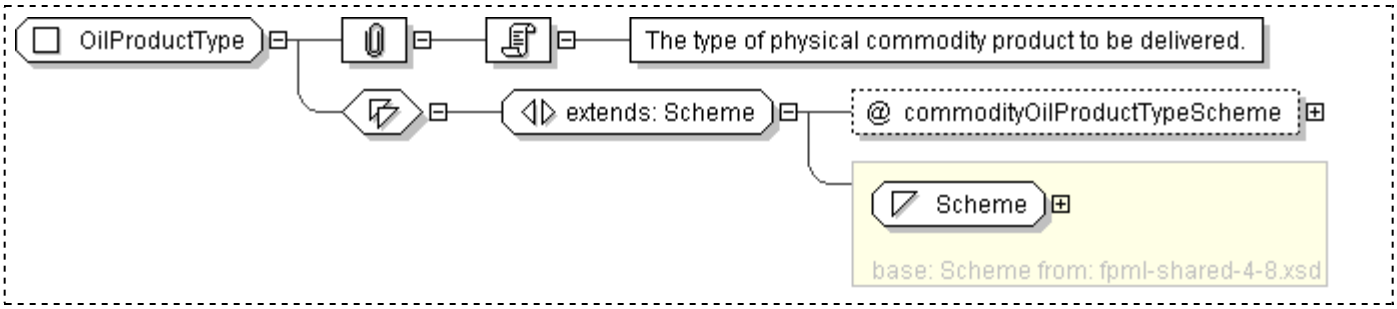
Super-types:	Scheme < OilProductType (by extension)
Sub-types:	None

Name	OilProductType
Used by (from the same schema document)	Complex Type OilProduct
Abstract	no
Documentation	The type of physical commodity product to be delivered.

XML Instance Representation

```
<...
 commodityOilProductTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilProductType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="commodityOilProductTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/commodity-oil-product-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OilTransferDelivery

[Table of contents]

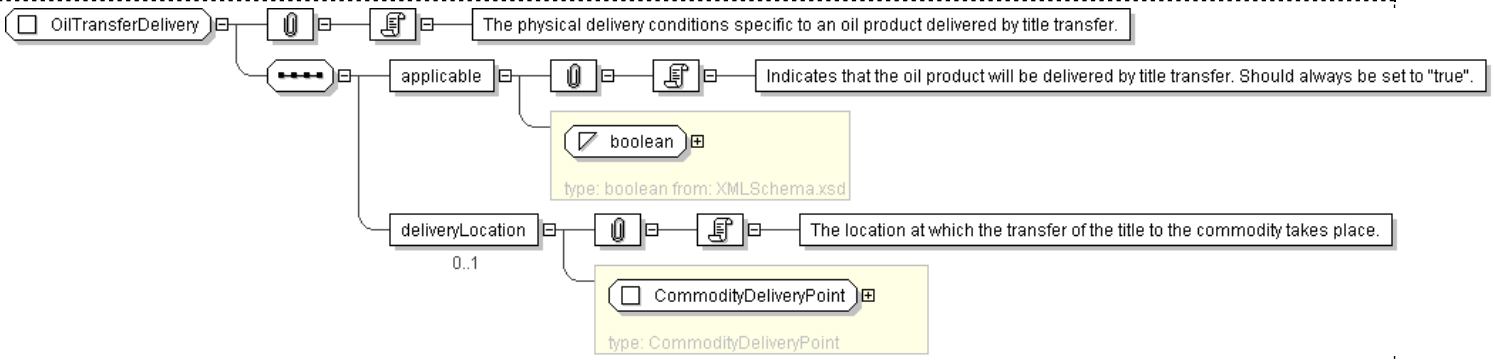
Super-types:	None
Sub-types:	None

Name	OilTransferDelivery
Used by (from the same schema document)	Complex Type OilDelivery
Abstract	no
Documentation	The physical delivery conditions specific to an oil product delivered by title transfer.

XML Instance Representation

```
<...>
  <applicable> xsd:boolean </applicable> [1]
  'Indicates that the oil product will be delivered by title transfer. Should always be set to \"true\".'

  <deliveryLocation> CommodityDeliveryPoint </deliveryLocation> [0..1]
  'The location at which the transfer of the title to the commodity takes place.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OilTransferDelivery">
  <xsd:sequence>
    <xsd:element name="applicable" type=" xsd:boolean "/>
    <xsd:element name="deliveryLocation" type=" CommodityDeliveryPoint " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PercentageTolerance

[Table of contents]

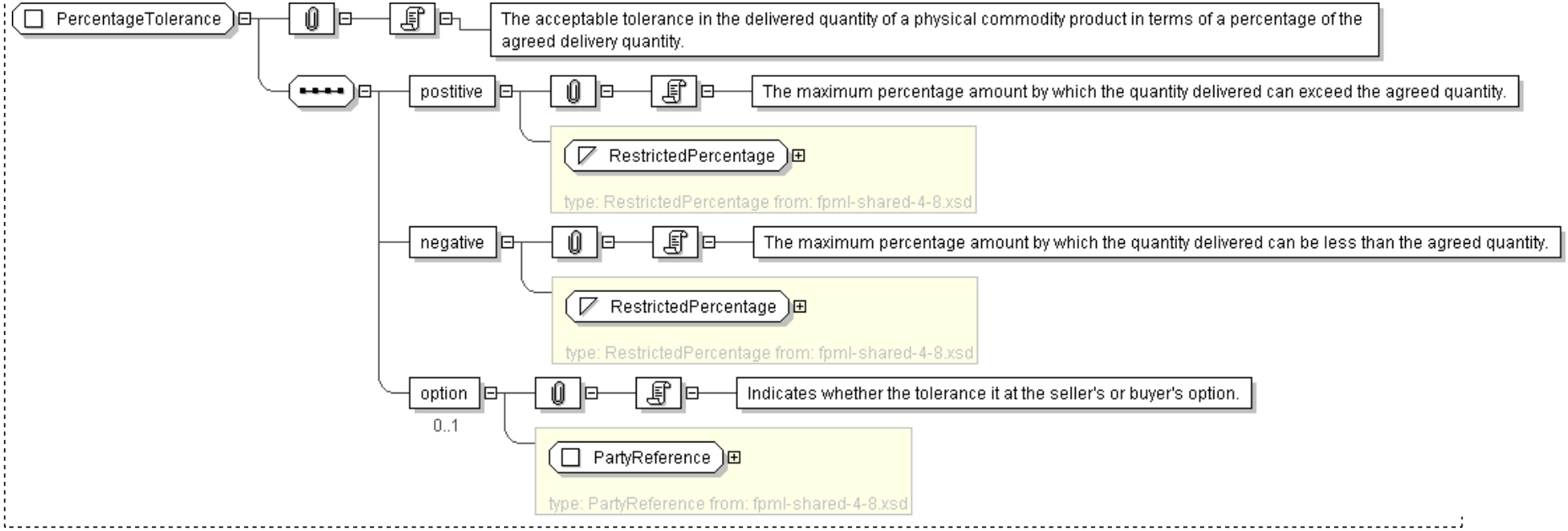
Super-types:	None
Sub-types:	None

Name	PercentageTolerance
Used by (from the same schema document)	Complex Type OilDelivery
Abstract	no
Documentation	The acceptable tolerance in the delivered quantity of a physical commodity product in terms of a percentage of the agreed delivery quantity.

XML Instance Representation

```
<...>
   <postitive> RestrictedPercentage </postitive> [1]
    'The maximum percentage amount by which the quantity delivered can exceed the agreed quantity.'

   <negative> RestrictedPercentage </negative> [1]
    'The maximum percentage amount by which the quantity delivered can be less than the agreed quantity.'

   <option> PartyReference </option> [0..1]
    'Indicates whether the tolerance it at the seller\'s or buyer\'s option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PercentageTolerance">
  <xsd:sequence>
    <xsd:element name="postitive" type=" RestrictedPercentage "/>
    <xsd:element name="negative" type=" RestrictedPercentage "/>
    <xsd:element name="option" type=" PartyReference " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PhysicalLeg

[Table of contents]

Super-types: Leg < PhysicalLeg (by extension)

Sub-types:

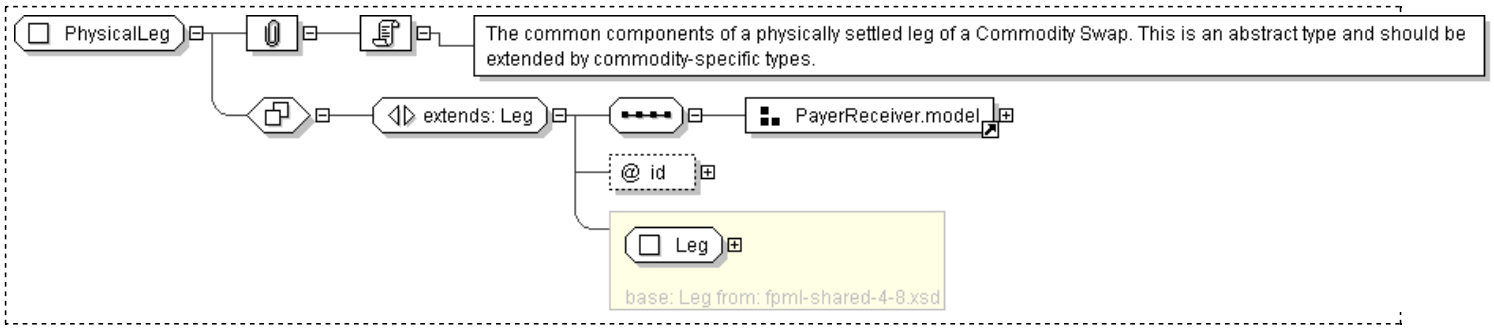
- BullionPhysicalLeg (by extension)
- CoalPhysicalLeg (by extension)
- ElectricityPhysicalLeg (by extension)
- GasPhysicalLeg (by extension)
- OilPhysicalLeg (by extension)

Name	PhysicalLeg
Abstract	yes
Documentation	The common components of a physically settled leg of a Commodity Swap. This is an abstract type and should be extended by commodity-specific types.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PhysicalLeg" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuantityReference

[Table of contents]

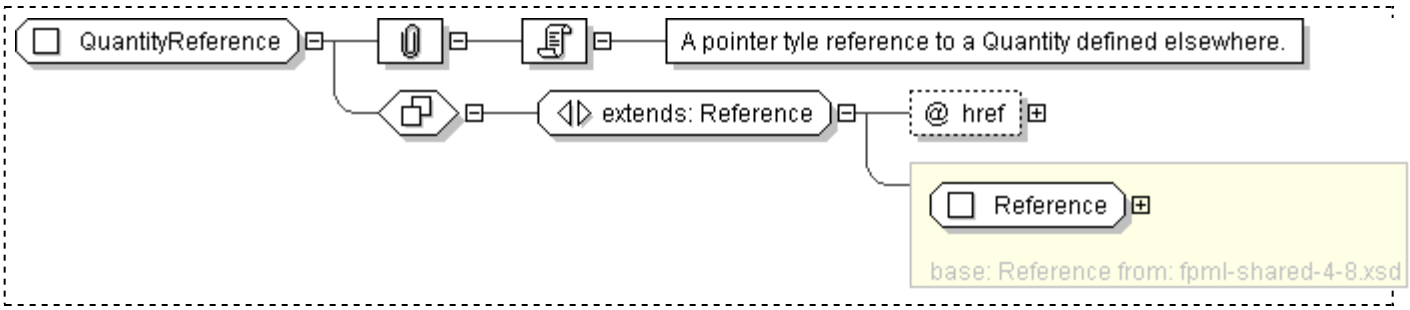
Super-types:	Reference < QuantityReference (by extension)
Sub-types:	None

Name	QuantityReference
Used by (from the same schema document)	Complex Type NonPeriodicFixedPriceLeg , Model Group CommodityNotionalQuantity.model
Abstract	no
Documentation	A pointer tyle reference to a Quantity defined elsewhere.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuantityReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="CommodityPhysicalQuantityBase"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuantityScheduleReference

[Table of contents]

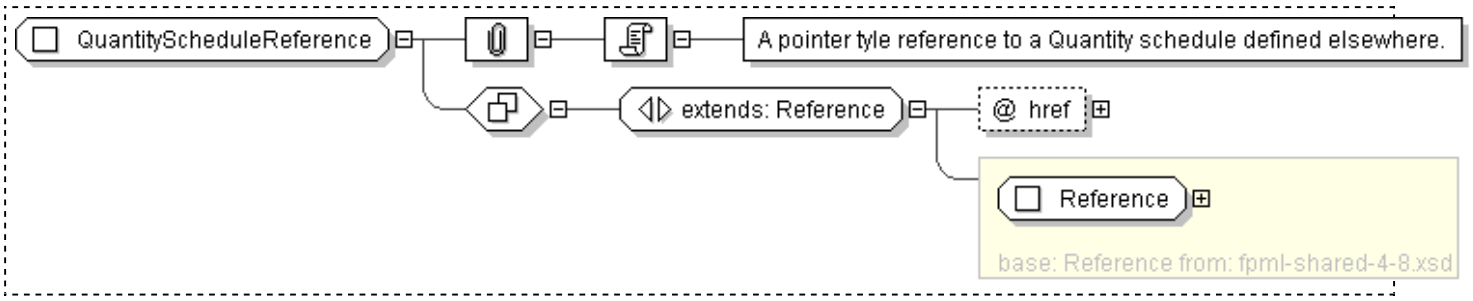
Super-types:	Reference < QuantityScheduleReference (by extension)
Sub-types:	None

Name	QuantityScheduleReference
Abstract	no
Documentation	A pointer tyle reference to a Quantity schedule defined elsewhere.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuantityScheduleReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="CommodityNotionalQuantitySchedule"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SequencedDisruptionFallback

[Table of contents]

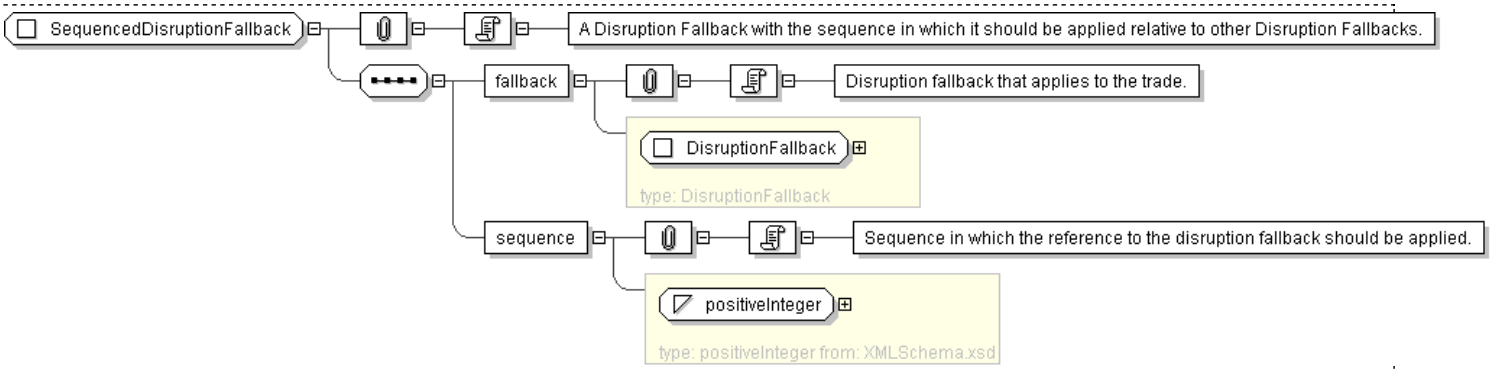
Super-types:	None
Sub-types:	None

Name	SequencedDisruptionFallback
Used by (from the same schema document)	Complex Type CommodityMarketDisruption
Abstract	no
Documentation	A Disruption Fallback with the sequence in which it should be applied relative to other Disruption Fallbacks.

XML Instance Representation

```
<...>
  <fallback> DisruptionFallback </fallback> [1]
   'Disruption fallback that applies to the trade.'

  <sequence> xsd:positiveInteger </sequence> [1]
   'Sequence in which the reference to the disruption fallback should be applied.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SequencedDisruptionFallback">
  <xsd:sequence>
    <xsd:element name="fallback" type=" DisruptionFallback "/>
    <xsd:element name="sequence" type=" xsd:positiveInteger "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPeriods

[Table of contents]

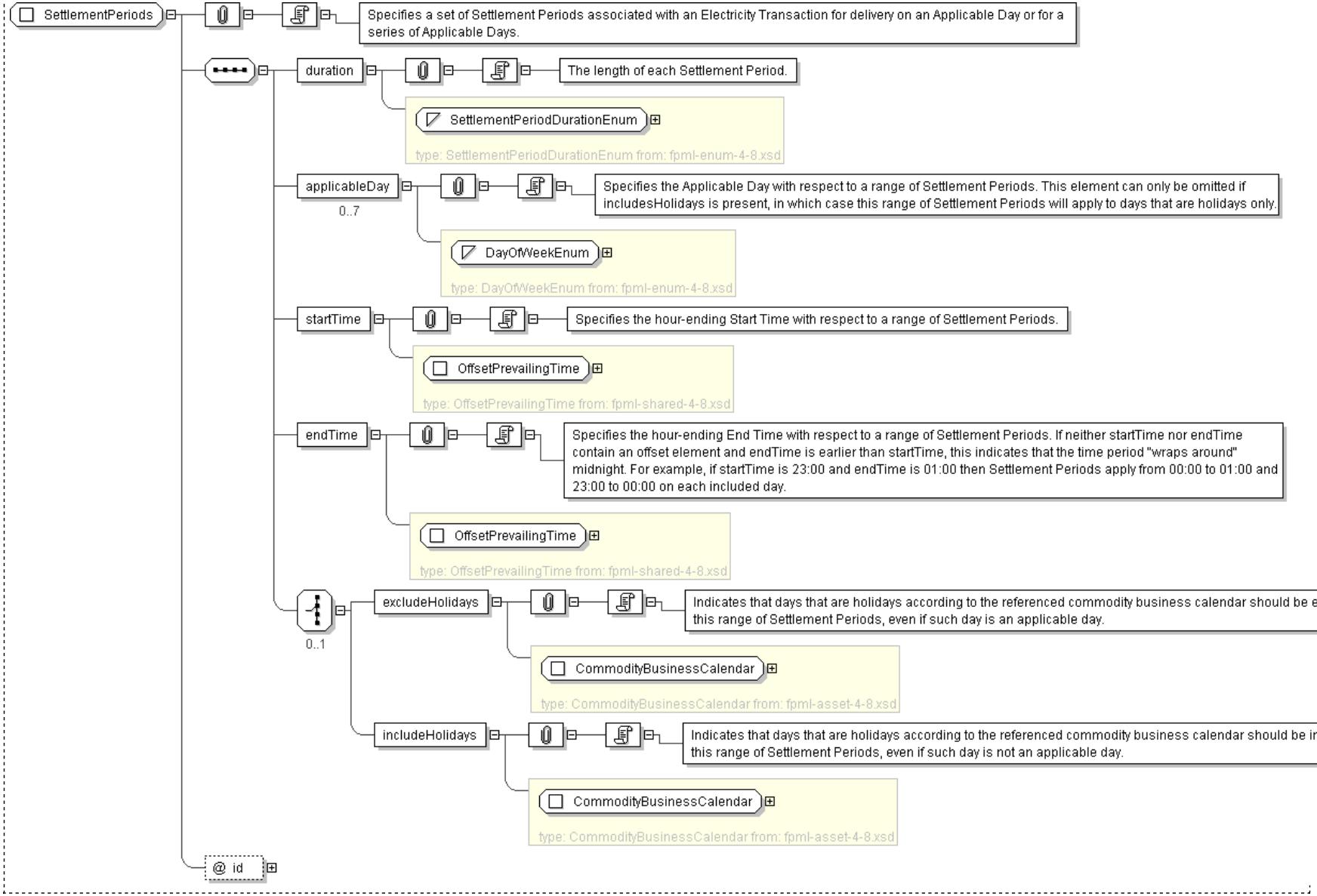
Super-types:	None
Sub-types:	None

Name	SettlementPeriods
Used by (from the same schema document)	Complex Type CommodityPricingDates , Complex Type ElectricityDeliveryPeriods , Complex Type ElectricityPhysicalLeg
Abstract	no
Documentation	Specifies a set of Settlement Periods associated with an Electricity Transaction for delivery on an Applicable Day or for a series of Applicable Days.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <duration> SettlementPeriodDurationEnum </duration> [1]
    'The length of each Settlement Period.'

   <applicableDay> DayOfWeekEnum </applicableDay> [0..7]
    'Specifies the Applicable Day with respect to a range of Settlement Periods. This element can only be omitted if includesHolidays is present, in
    which case this range of Settlement Periods will apply to days that are holidays only.'

   <startTime> OffsetPrevailingTime </startTime> [1]
    'Specifies the hour-ending Start Time with respect to a range of Settlement Periods.'

   <endTime> OffsetPrevailingTime </endTime> [1]
    'Specifies the hour-ending End Time with respect to a range of Settlement Periods. If neither startTime nor endTime contain an offset element and
    endTime is earlier than startTime, this indicates that the time period \"wraps around\" midnight. For example, if startTime is 23:00 and endTime
    is 01:00 then Settlement Periods apply from 00:00 to 01:00 and 23:00 to 00:00 on each included day.'

   Start Choice [0..1]
      <excludeHolidays> CommodityBusinessCalendar </excludeHolidays> [1]
       'Indicates that days that are holidays according to the referenced commodity business calendar should be excluded from this range of Settlement
       Periods, even if such day is an applicable day.'

      <includeHolidays> CommodityBusinessCalendar </includeHolidays> [1]
       'Indicates that days that are holidays according to the referenced commodity business calendar should be included in this range of Settlement
       Periods, even if such day is not an applicable day.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPeriods">
  <xsd:sequence>
    <xsd:element name="duration" type=" SettlementPeriodDurationEnum "/>
    <xsd:element name="applicableDay" type=" DayOfWeekEnum " minOccurs="0" maxOccurs="7"/>
    <xsd:element name="startTime" type=" OffsetPrevailingTime "/>
    <xsd:element name="endTime" type=" OffsetPrevailingTime "/>
    <xsd:choice minOccurs="0">
      <xsd:element name="excludeHolidays" type=" CommodityBusinessCalendar "/>
      <xsd:element name="includeHolidays" type=" CommodityBusinessCalendar "/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: SettlementPeriodsFixedPrice

[Table of contents]

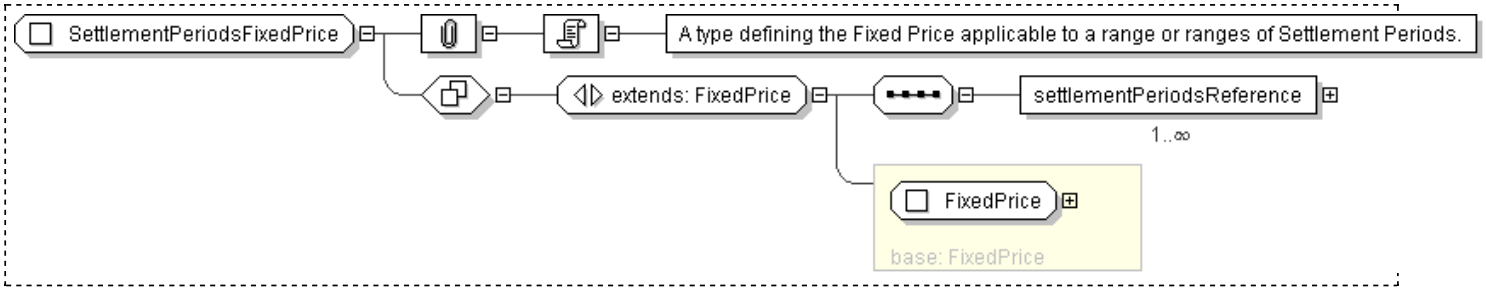
Super-types:	FixedPrice < SettlementPeriodsFixedPrice (by extension)
Sub-types:	None

Name	SettlementPeriodsFixedPrice
Used by (from the same schema document)	Model Group CommodityFixedPrice.model
Abstract	no
Documentation	A type defining the Fixed Price applicable to a range or ranges of Settlement Periods.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <price> xsd:decimal </price> [1]
   'The Fixed Price.'

   <priceCurrency> Currency </priceCurrency> [1]
   'Currency of the fixed price.'

   <priceUnit> QuantityUnit </priceUnit> [1]
   'The unit of measure used to calculate the Fixed Price.'

   <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPeriodsFixedPrice">
  <xsd:complexContent>
    <xsd:extension base=" FixedPrice ">
      <xsd:sequence>
        <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference "
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPeriodsReference

[Table of contents]

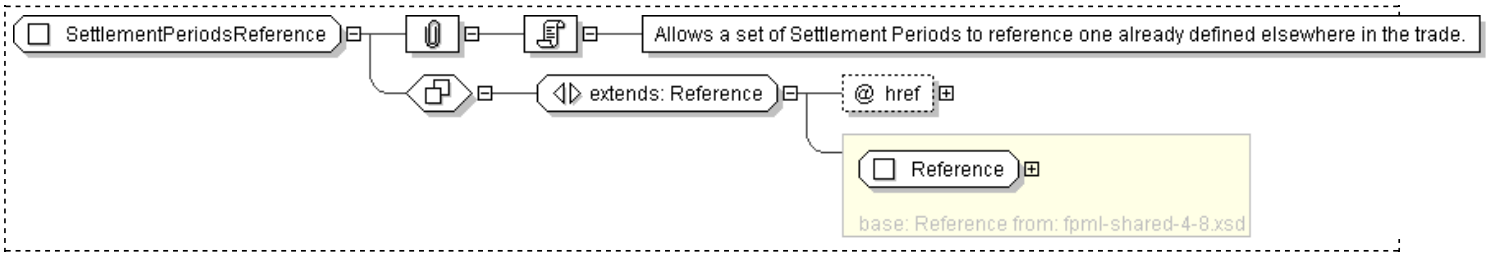
Super-types:	Reference < SettlementPeriodsReference (by extension)
Sub-types:	None

Name	SettlementPeriodsReference
Used by (from the same schema document)	Complex Type CommodityPricingDates , Complex Type CommoditySettlementPeriodsNotionalQuantity , Complex Type CommoditySettlementPeriodsNotionalQuantitySchedule , Complex Type CommoditySettlementPeriodsPriceSchedule , Complex Type ElectricityPhysicalDeliveryQuantity , Complex Type ElectricityPhysicalDeliveryQuantitySchedule , Complex Type SettlementPeriodsFixedPrice , Complex Type SettlementPeriodsStep
Abstract	no
Documentation	Allows a set of Settlement Periods to reference one already defined elsewhere in the trade.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPeriodsReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="SettlementPeriods"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPeriodsSchedule

[Table of contents]

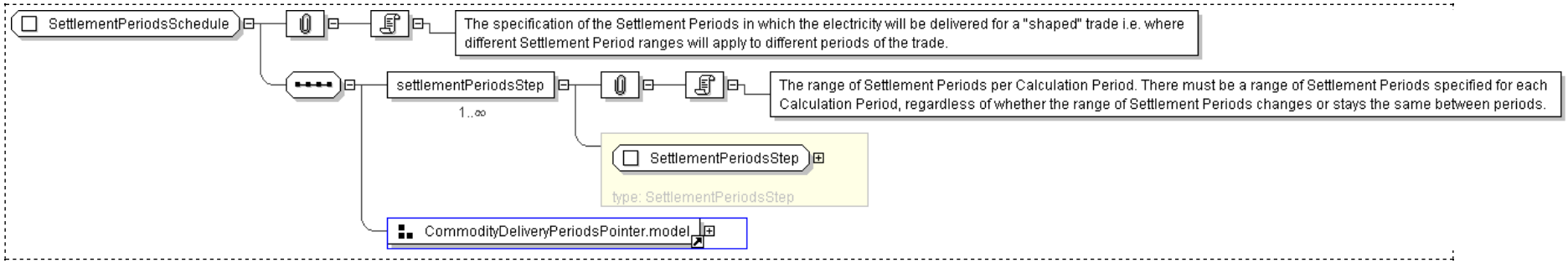
Super-types:	None
Sub-types:	None

Name	SettlementPeriodsSchedule
Used by (from the same schema document)	Complex Type ElectricityPhysicalLeg
Abstract	no
Documentation	The specification of the Settlement Periods in which the electricity will be delivered for a "shaped" trade i.e. where different Settlement Period ranges will apply to different periods of the trade.

XML Instance Representation

```
<...>
  <settlementPeriodsStep> SettlementPeriodsStep </settlementPeriodsStep> [1..*]
   'The range of Settlement Periods per Calculation Period. There must be a range of Settlement Periods specified for each Calculation Period,
   regardless of whether the range of Settlement Periods changes or stays the same between periods.'

  Start Choice [1]
    <deliveryPeriodsReference> CalculationPeriodsReference </deliveryPeriodsReference> [1]
     'A pointer style reference to the Delivery Periods defined elsewhere.'

    <deliveryPeriodsScheduleReference> CalculationPeriodsScheduleReference </deliveryPeriodsScheduleReference> [1]
     'A pointer style reference to the Calculation Periods Schedule defined elsewhere.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPeriodsSchedule">
  <xsd:sequence>
    <xsd:element name="settlementPeriodsStep" type=" SettlementPeriodsStep " maxOccurs="unbounded"/>
    <xsd:group ref=" CommodityDeliveryPeriodsPointer.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPeriodsStep

[Table of contents]

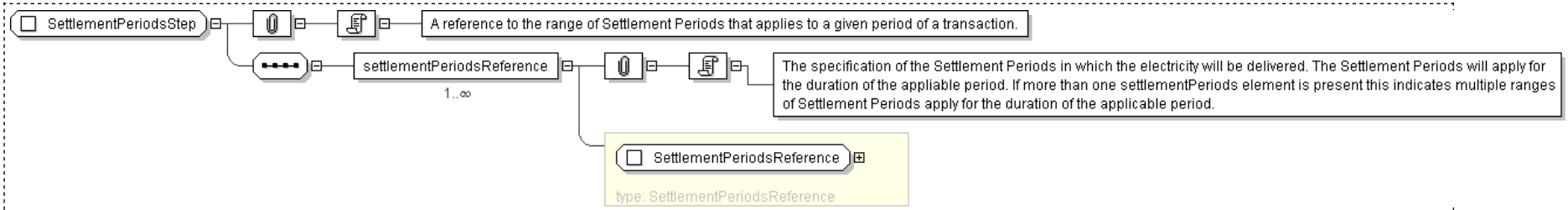
Super-types:	None
Sub-types:	None

Name	SettlementPeriodsStep
Used by (from the same schema document)	Complex Type SettlementPeriodsSchedule
Abstract	no
Documentation	A reference to the range of Settlement Periods that applies to a given period of a transaction.

XML Instance Representation

```
<...>
  <settlementPeriodsReference> SettlementPeriodsReference </settlementPeriodsReference> [1..*]
  'The specification of the Settlement Periods in which the electricity will be delivered. The Settlement Periods will apply for the duration of the
  appliable period. If more than one settlementPeriods element is present this indicates multiple ranges of Settlement Periods apply for the duration
  of the applicable period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPeriodsStep">
  <xsd:sequence>
    <xsd:element name="settlementPeriodsReference" type=" SettlementPeriodsReference " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: UnitQuantity

[Table of contents]

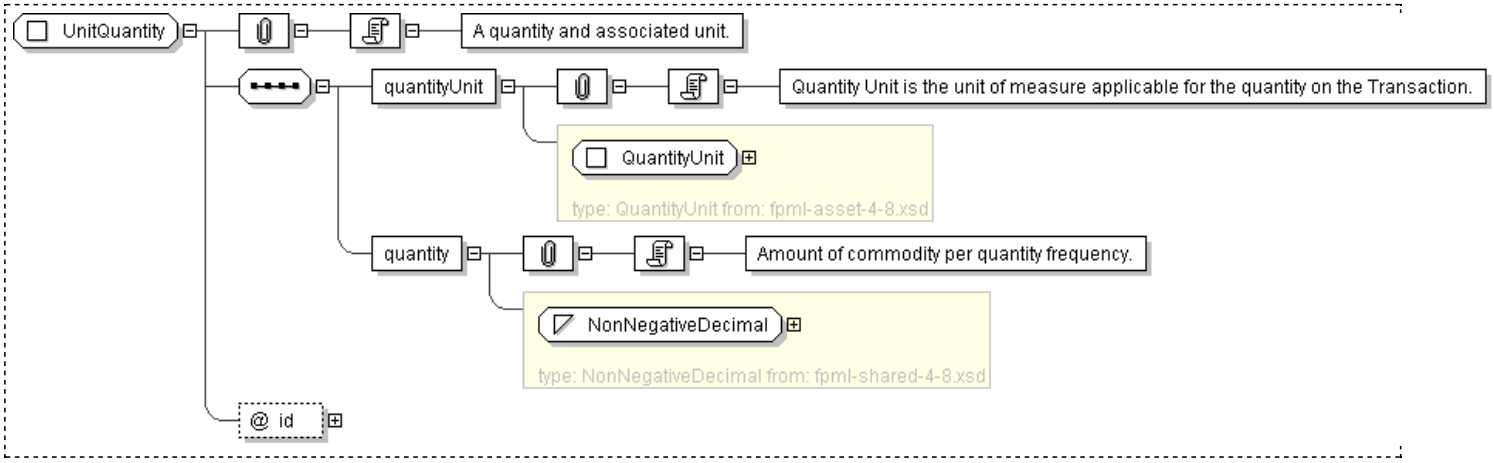
Super-types:	None
Sub-types:	None

Name	UnitQuantity
Used by (from the same schema document)	Complex Type ElectricityPhysicalQuantity , Model Group CommodityFixedPhysicalQuantity.model
Abstract	no
Documentation	A quantity and associated unit.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <quantityUnit> QuantityUnit </quantityUnit> [1]
    'Quantity Unit is the unit of measure applicable for the quantity on the Transaction.'

   <quantity> NonNegativeDecimal </quantity> [1]
    'Amount of commodity per quantity frequency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="UnitQuantity">
  <xsd:sequence>
    <xsd:element name="quantityUnit" type=" QuantityUnit "/>
    <xsd:element name="quantity" type=" NonNegativeDecimal "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: CancelTradeConfirmation
  - Complex Type: ConfirmTrade
  - Complex Type: ConfirmationCancelled
  - Complex Type: ModifyTradeConfirmation
  - Complex Type: RequestTradeConfirmation
  - Complex Type: TradeAffirmation
  - Complex Type: TradeAffirmed
  - Complex Type: TradeAlreadyAffirmed
  - Complex Type: TradeAlreadyConfirmed
  - Complex Type: TradeConfirmed
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
```

```
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the

substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancelTradeConfirmation

[Table of contents]

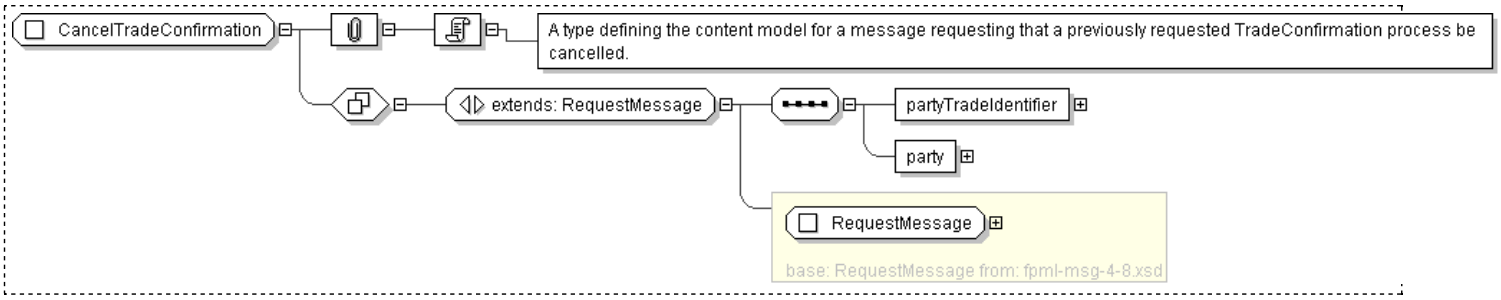
Super-types:	RequestMessage < CancelTradeConfirmation (by extension)
Sub-types:	None

Name	CancelTradeConfirmation
Abstract	no
Documentation	A type defining the content model for a message requesting that a previously requested TradeConfirmation process be cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead,
 it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build
 number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a
 version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases
 within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1]
   'The trade reference identifier(s) allocated to the trade by the parties involved.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For
   example, the principal parties obligated to make payments from time to time during the term of the trade, but may
   include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation
   transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a
   document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancelTradeConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ConfirmationCancelled

[Table of contents]

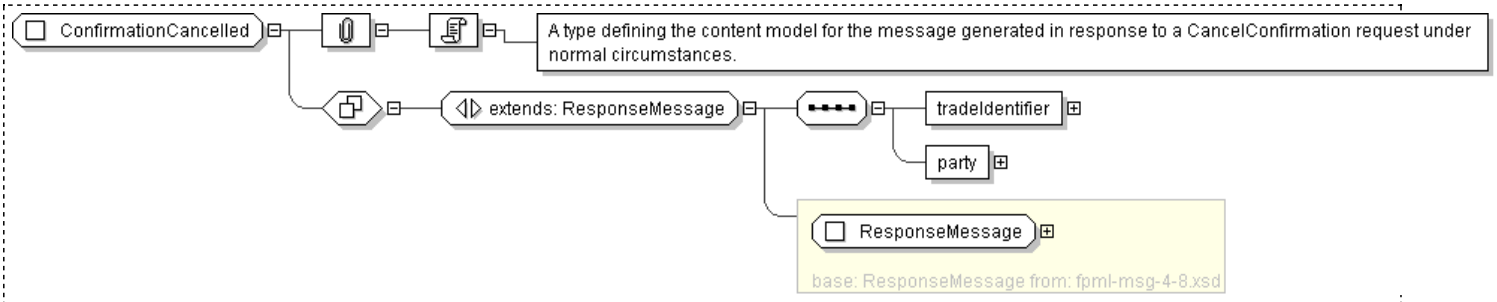
Super-types:	ResponseMessage < ConfirmationCancelled (by extension)
Sub-types:	None

Name	ConfirmationCancelled
Abstract	no
Documentation	A type defining the content model for the message generated in response to a CancelConfirmation request under normal circumstances.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle.
   For example, the principal parties obligated to make payments from time to time during the term of the trade,
   but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of
   novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places
   within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ConfirmationCancelled">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ConfirmTrade

[Table of contents]

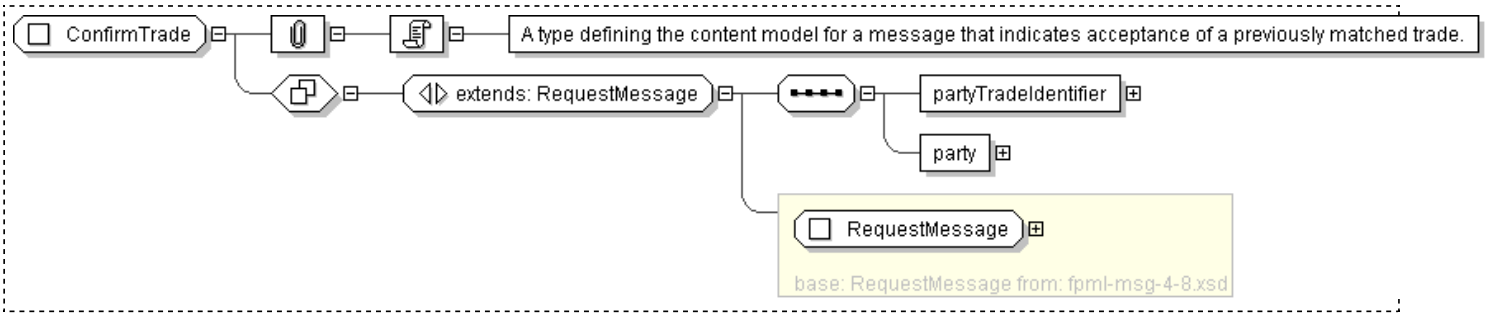
Super-types:	RequestMessage < ConfirmTrade (by extension)
Sub-types:	None

Name	ConfirmTrade
Abstract	no
Documentation	A type defining the content model for a message that indicates acceptance of a previously matched trade.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1]
   'The trade reference identifier(s) allocated to the trade by the parties involved.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a
   trade lifecycle. For example, the principal parties obligated to make payments from time to time
   during the term of the trade, but may include other parties involved in, or incidental to, the
   trade, such as parties acting in the role of novation transferor/transferee, broker, calculation
   agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ConfirmTrade">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ModifyTradeConfirmation

[Table of contents]

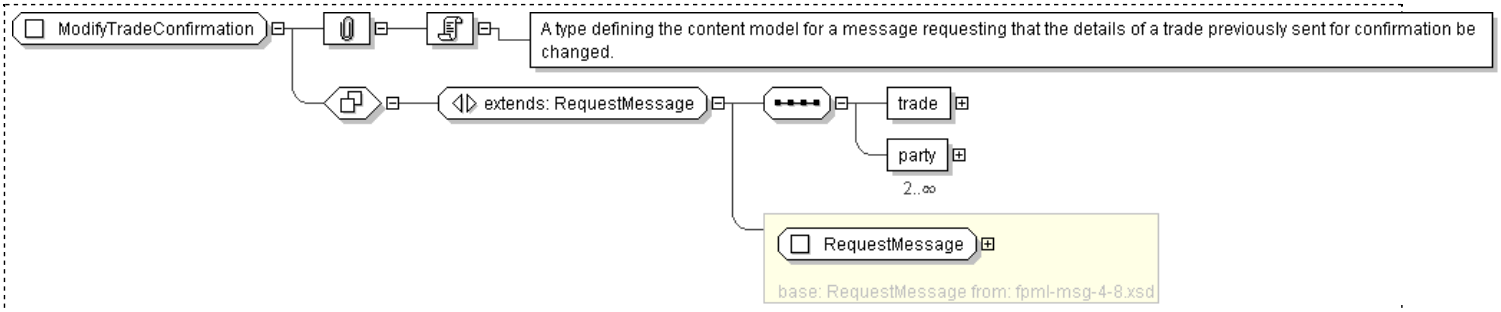
Super-types:	RequestMessage < ModifyTradeConfirmation (by extension)
Sub-types:	None

Name	ModifyTradeConfirmation
Abstract	no
Documentation	A type defining the content model for a message requesting that the details of a trade previously sent for confirmation be changed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead,
 it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build
 number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a
 version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases
 within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For
   example, the principal parties obligated to make payments from time to time during the term of the trade, but may
   include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation
   transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a
   document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ModifyTradeConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestTradeConfirmation

[Table of contents]

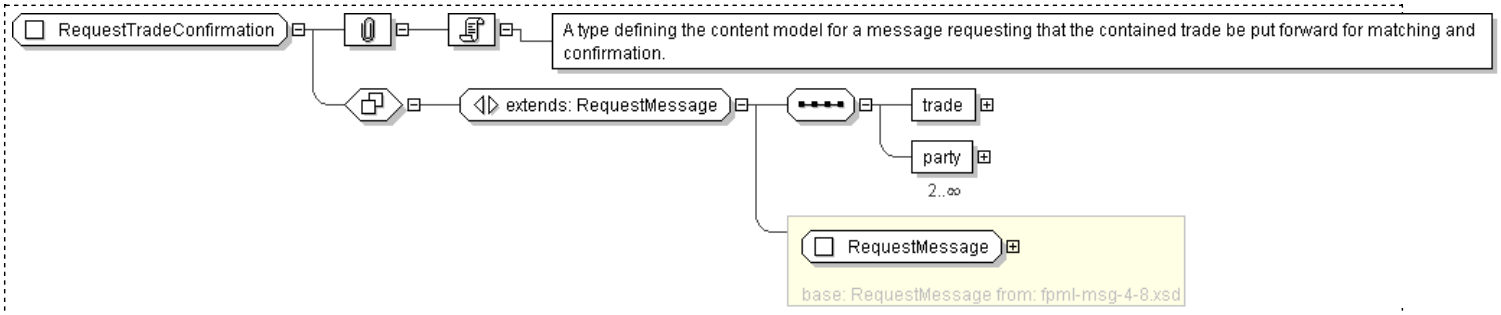
Super-types:	RequestMessage < RequestTradeConfirmation (by extension)
Sub-types:	None

Name	RequestTradeConfirmation
Abstract	no
Documentation	A type defining the content model for a message requesting that the contained trade be put forward for matching and confirmation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead,
 it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build
 number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a
 version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases
 within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For
   example, the principal parties obligated to make payments from time to time during the term of the trade, but may
   include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation
   transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a
   document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestTradeConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAffirmation

[Table of contents]

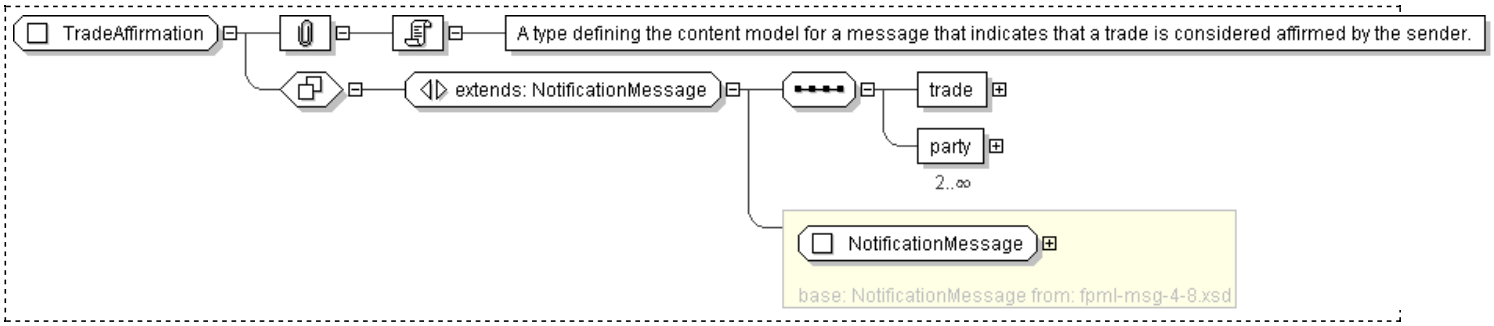
Super-types:	NotificationMessage < TradeAffirmation (by extension)
Sub-types:	None

Name	TradeAffirmation
Abstract	no
Documentation	A type defining the content model for a message that indicates that a trade is considered affirmed by the sender.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema,
 validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented.
 If no changes have been made between releases within a version (i.e. from Trial Recommendation to
 Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the
   term of the trade, but may include other parties involved in, or incidental to, the trade, such as
   parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML
   roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAffirmation">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAffirmed

[Table of contents]

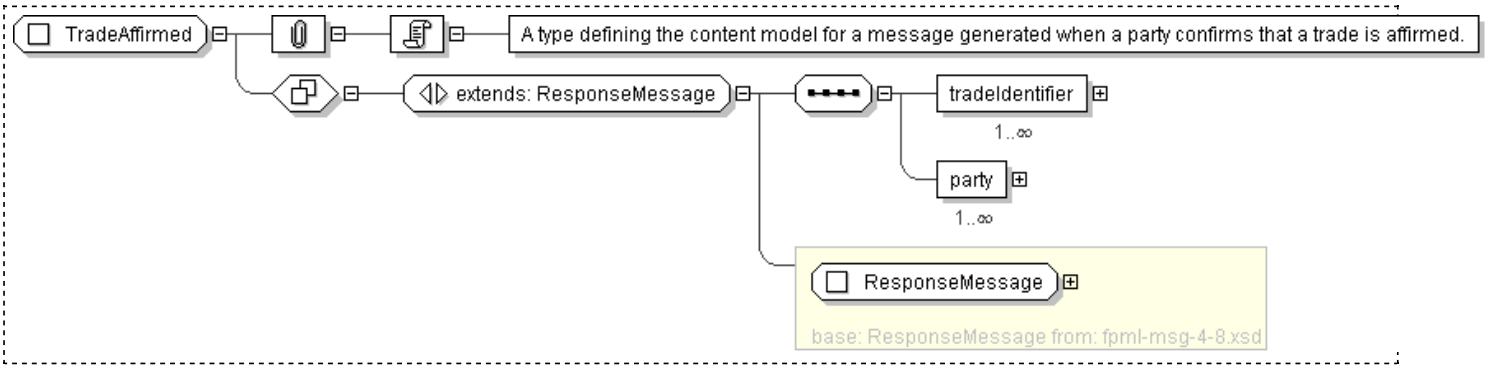
Super-types:	ResponseMessage < TradeAffirmed (by extension)
Sub-types:	None

Name	TradeAffirmed
Abstract	no
Documentation	A type defining the content model for a message generated when a party confirms that a trade is affirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1..*]
   'An instance of a unique trade identifier.'

   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during
   the term of the trade, but may include other parties involved in, or incidental to, the trade,
   such as parties acting in the role of novation transferor/transferee, broker, calculation agent,
   etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAffirmed">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadyAffirmed

[Table of contents]

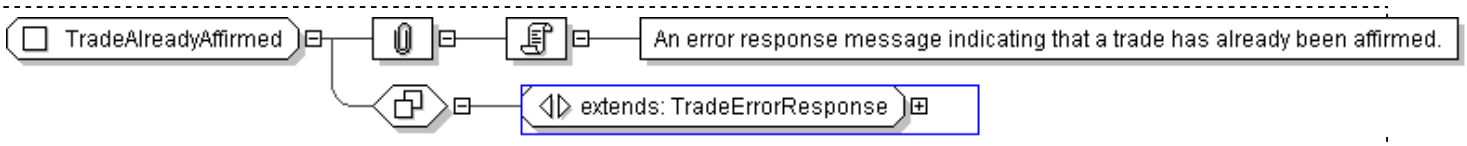
Super-types:	TradeErrorResponse < TradeAlreadyAffirmed (by extension)
Sub-types:	None

Name	TradeAlreadyAffirmed
Abstract	no
Documentation	An error response message indicating that a trade has already been affirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different party.'

   End Choice
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadyAffirmed">
```

```
   <xsd:complexContent>
      <xsd:extension base=" TradeErrorResponse "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadyConfirmed

[Table of contents]

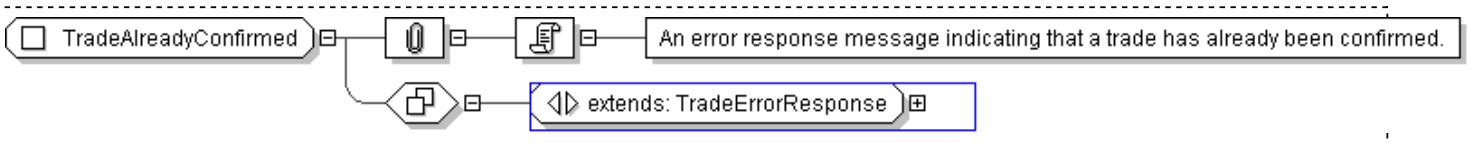
Super-types:	TradeErrorResponse < TradeAlreadyConfirmed (by extension)
Sub-types:	None

Name	TradeAlreadyConfirmed
Abstract	no
Documentation	An error response message indicating that a trade has already been confirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different party.'

   End Choice
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadyConfirmed">
   <xsd:complexContent>
      <xsd:extension base=" TradeErrorResponse "/>
   </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeConfirmed

[Table of contents]

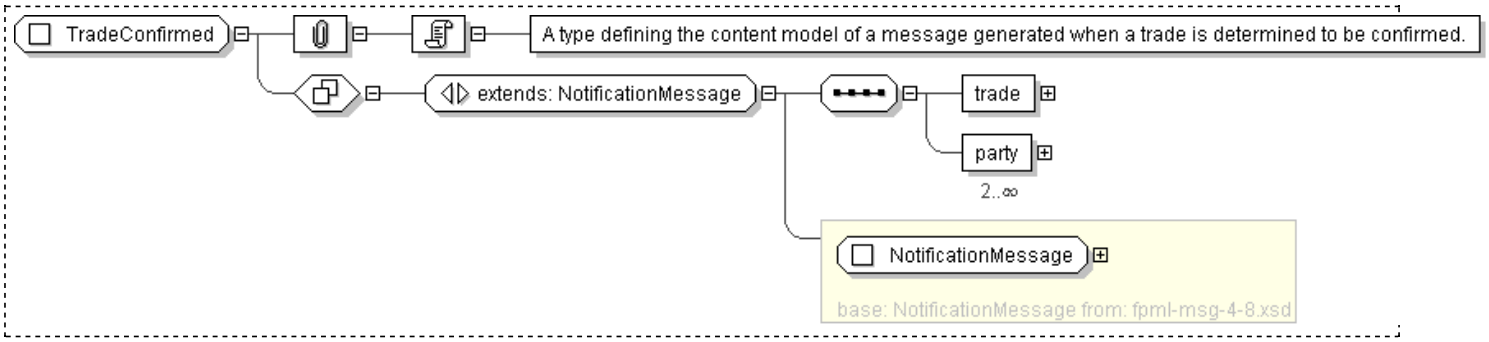
Super-types:	NotificationMessage < TradeConfirmed (by extension)
Sub-types:	None

Name	TradeConfirmed
Abstract	no
Documentation	A type defining the content model of a message generated when a trade is determined to be confirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during
   the term of the trade, but may include other parties involved in, or incidental to, the trade, such
   as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In
   FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeConfirmed">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: changeEvent
  - Element: indexChange
- Global Definitions
  - Complex Type: ChangeEvent
  - Complex Type: ContractAmended
  - Complex Type: ContractAmendedCancelled
  - Complex Type: ContractAmendment
  - Complex Type: ContractCancelled
  - Complex Type: ContractChange
  - Complex Type: ContractChanged
  - Complex Type: ContractChangedCancelled
  - Complex Type: ContractCreated
  - Complex Type: ContractFullTermination
  - Complex Type: ContractFullTerminationCancelled
  - Complex Type: ContractIncreased
  - Complex Type: ContractIncreasedCancelled
  - Complex Type: ContractNovated
  - Complex Type: ContractNovatedCancelled
  - Complex Type: ContractPartialTermination
  - Complex Type: ContractPartialTerminationCancelled
  - Complex Type: ContractReferenceMessage
  - Complex Type: IndexChange
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ◦ fpml-msg-4-8.xsd
	  ◦ fpml-reconciliation-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace

ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-reconciliation-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* =

[1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

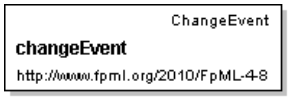
Element: changeEvent

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - indexChange

Name	changeEvent
Used by (from the same schema document)	Complex Type ContractChange
Type	ChangeEvent
Nillable	no
Abstract	yes
Documentation	The head of the substitution group for all elements which define the changes resulting from a market event.

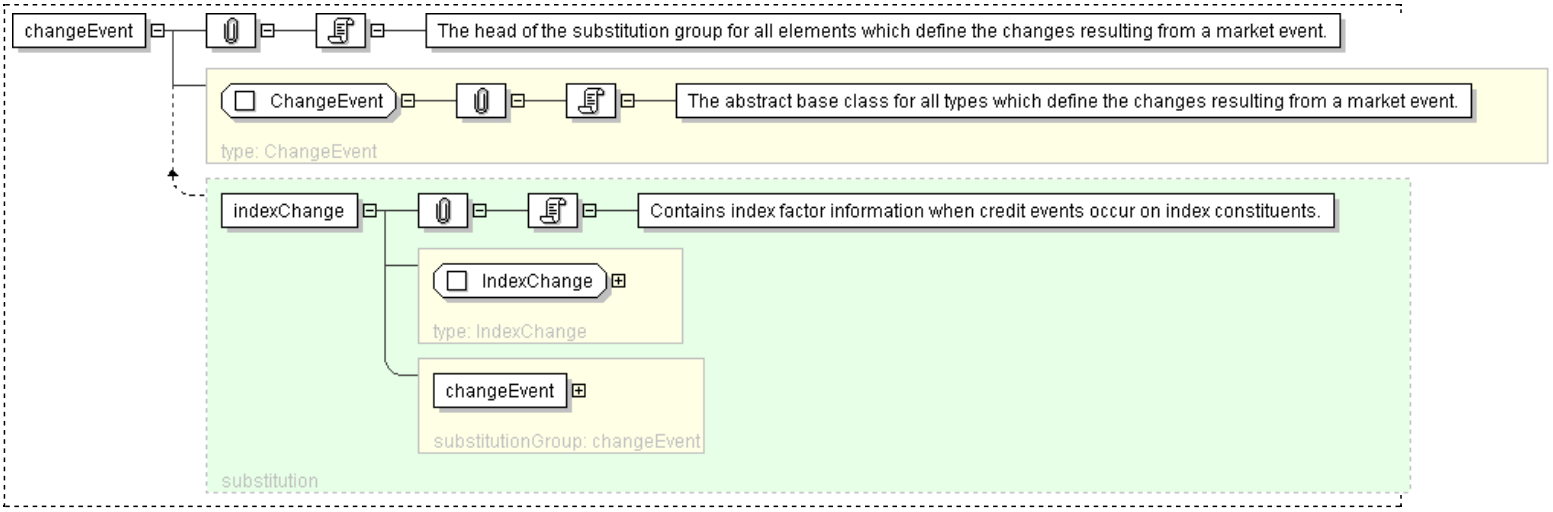
Logical Diagram



XML Instance Representation

```
<changeEvent/>
```

Diagram



Schema Component Representation

```
<xsd:element name="changeEvent" type=" ChangeEvent " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

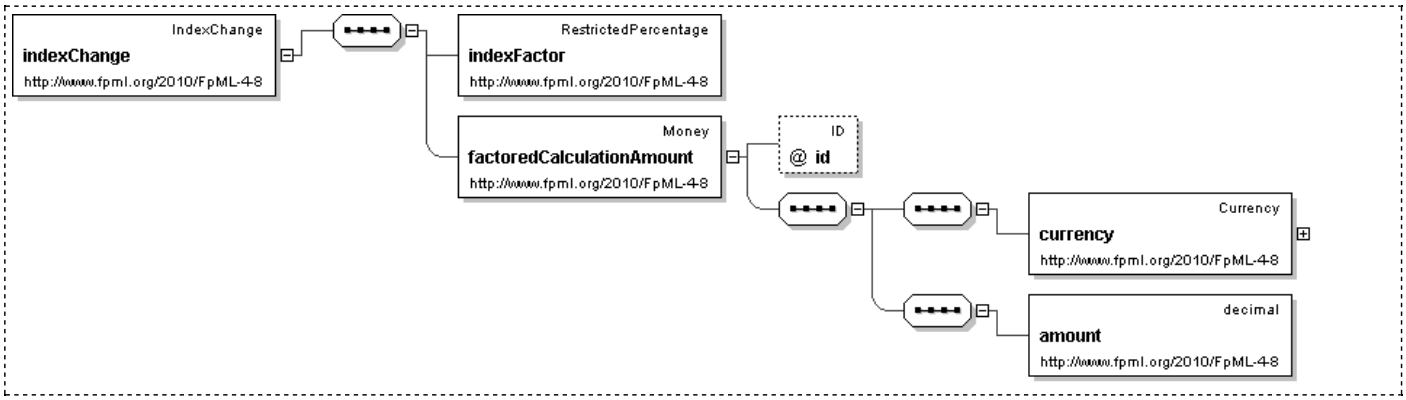
Element: indexChange

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - changeEvent

Name	indexChange
Type	IndexChange
Nillable	no
Abstract	no
Documentation	Contains index factor information when credit events occur on index constituents.

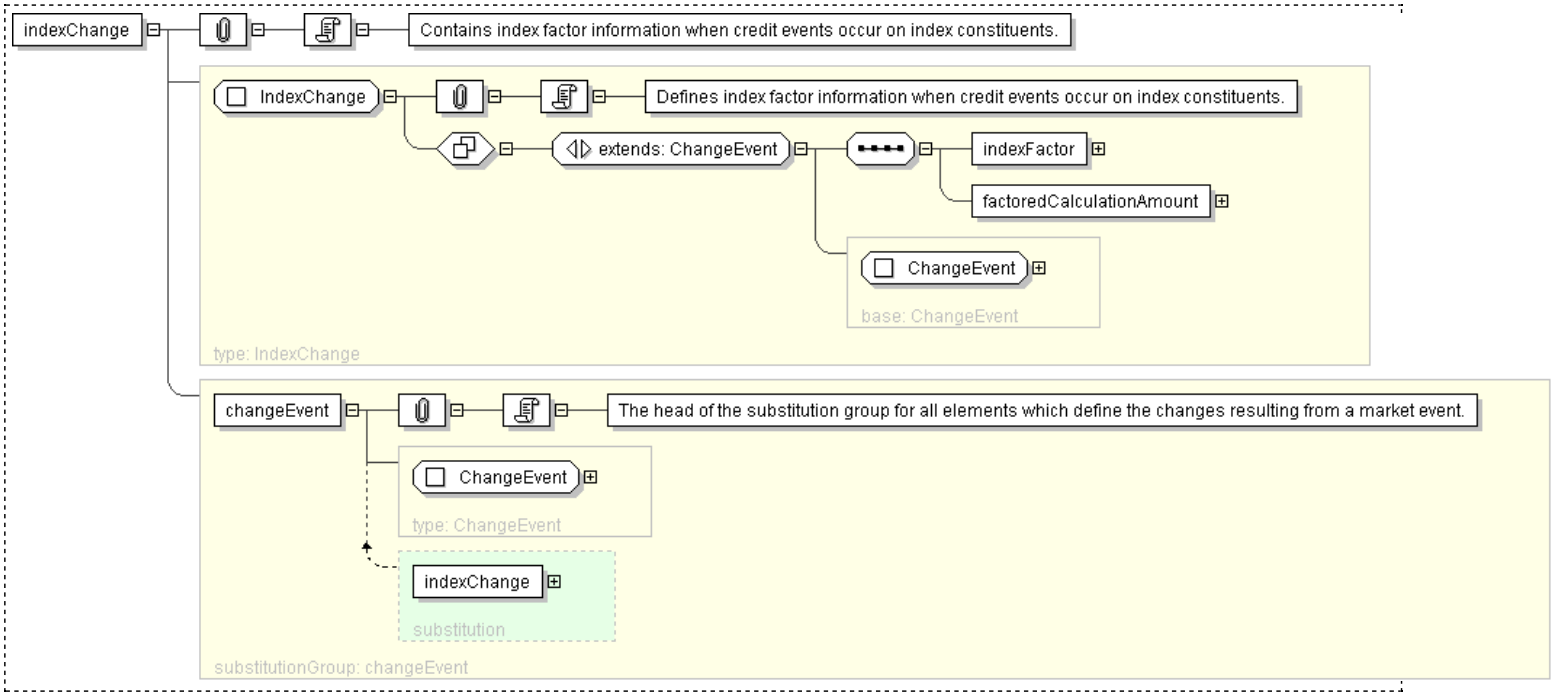
Logical Diagram



XML Instance Representation

```
<indexChange>
   <indexFactor> RestrictedPercentage </indexFactor> [1]
   'The index version factor, expressed as a decimal, that multiplied by the original notional amount yields the
   notional amount covered by the seller. A factor of 75% would be represented as 0.75.'

   <factoredCalculationAmount> Money </factoredCalculationAmount> [1]
   'This relates only to CDS index contracts and represents the factored notional amount, obtained by multiplying
   the index factor to the contract notional amount. See indexFactor description above.'

</indexChange>
```

Diagram



Schema Component Representation

```
<xsd:element name="indexChange" type=" IndexChange " substitutionGroup="changeEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ChangeEvent

[Table of contents]

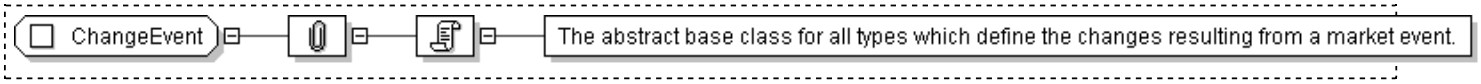
Super-types:	None
Sub-types:	• IndexChange (by extension)

Name	ChangeEvent
Used by (from the same schema document)	Element changeEvent
Abstract	yes
Documentation	The abstract base class for all types which define the changes resulting from a market event.

XML Instance Representation

```
<.../>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ChangeEvent" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractAmended

[Table of contents]

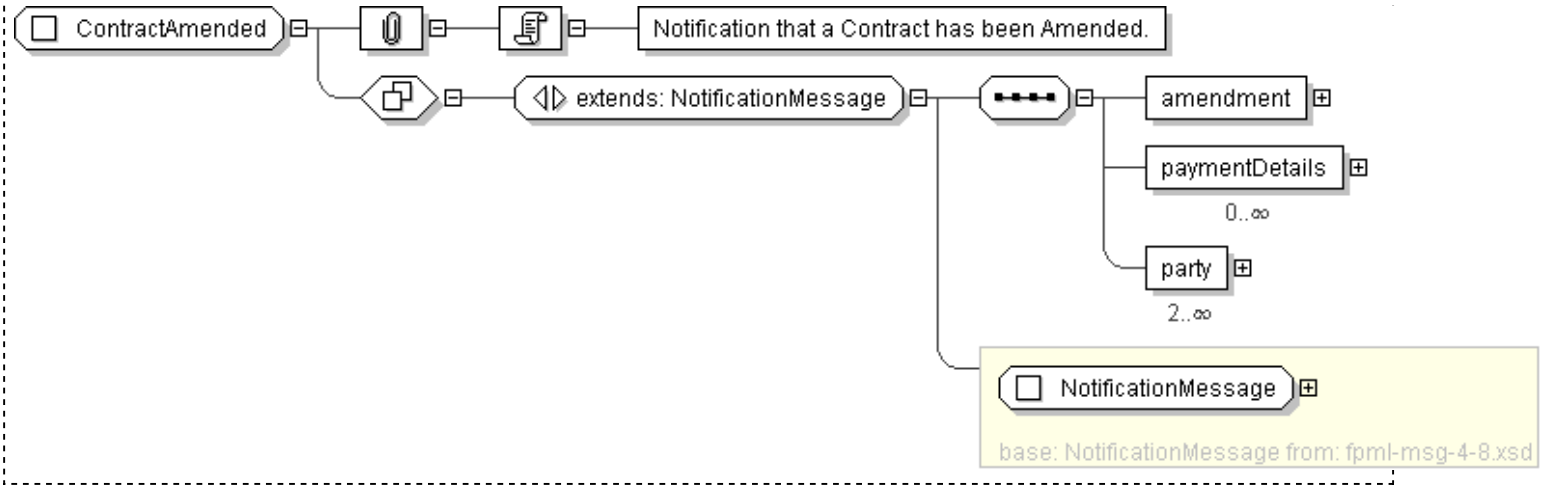
Super-types:	NotificationMessage < ContractAmended (by extension)
Sub-types:	None

Name	ContractAmended
Abstract	no
Documentation	Notification that a Contract has been Amended.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> ContractAmendment </amendment> [1]
   'Details of the amendment'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractAmended">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="amendment" type=" ContractAmendment "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractAmendedCancelled

[Table of contents]

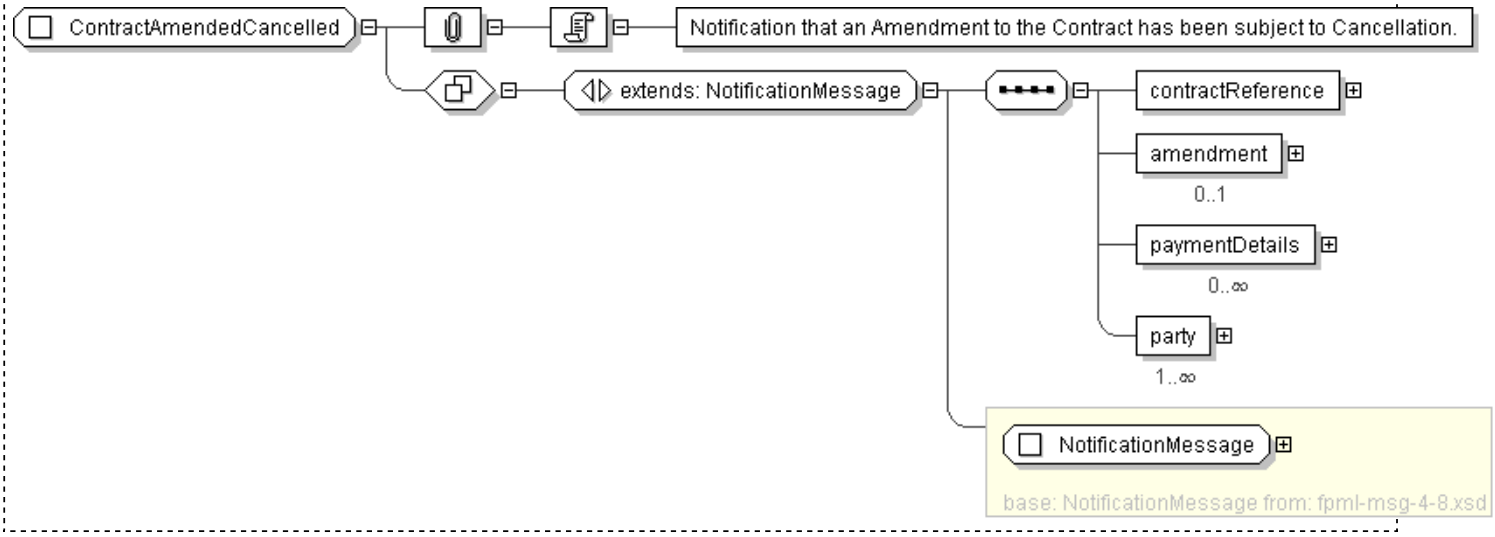
Super-types:	NotificationMessage < ContractAmendedCancelled (by extension)
Sub-types:	None

Name	ContractAmendedCancelled
Abstract	no
Documentation	Notification that an Amendment to the Contract has been subject to Cancellation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Identification of a single Contract which is the subject of this message. Each Party may provide one to many identifiers for this contract.'

   <amendment> ContractAmendment </amendment> [0..1]
   'Details of the amendment being cancelled'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractAmendedCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="amendment" type=" ContractAmendment " minOccurs="0"/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractAmendment

[Table of contents]

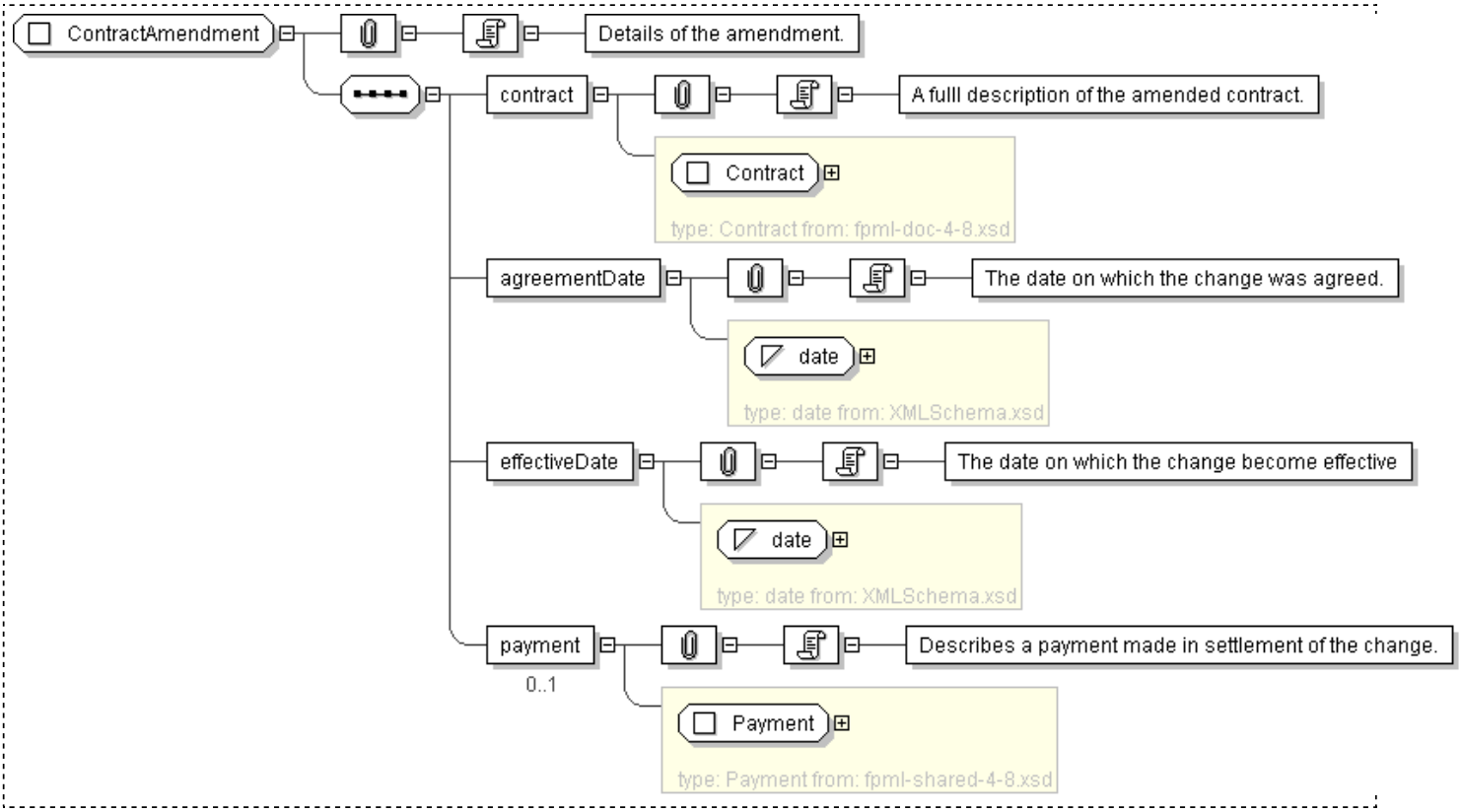
Super-types:	None
Sub-types:	None

Name	ContractAmendment
Used by (from the same schema document)	Complex Type ContractAmended , Complex Type ContractAmendedCancelled
Abstract	no
Documentation	Details of the amendment.

XML Instance Representation

```
<...>
   <contract> Contract </contract> [1]
   'A fulll description of the amended contract.'

   <agreementDate> xsd:date </agreementDate> [1]
   'The date on which the change was agreed.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the change become effective'

   <payment> Payment </payment> [0..1]
   'Describes a payment made in settlement of the change.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractAmendment">
   <xsd:sequence>
```

```
      <xsd:element name="contract" type=" Contract "/>
      <xsd:element name="agreementDate" type=" xsd:date "/>
      <xsd:element name="effectiveDate" type=" xsd:date "/>
      <xsd:element name="payment" type=" Payment " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractCancelled

[Table of contents]

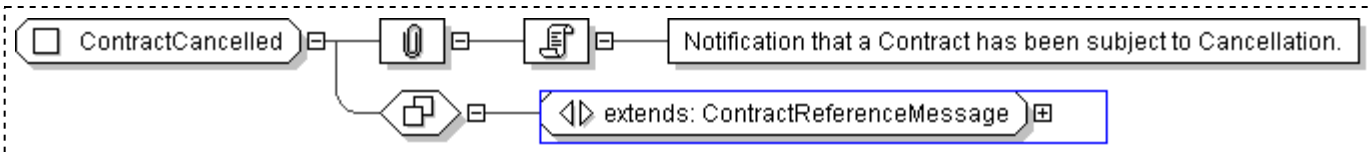
Super-types:	NotificationMessage < ContractReferenceMessage (by extension) < ContractCancelled (by extension)
Sub-types:	None

Name	ContractCancelled
Abstract	no
Documentation	Notification that a Contract has been subject to Cancellation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Identification of a single Contract which is the subject of this message. Each Party may provide one to many identifiers for this contract.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractCancelled">
   <xsd:complexContent>
      <xsd:extension base=" ContractReferenceMessage "/>
   </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractChange

[Table of contents]

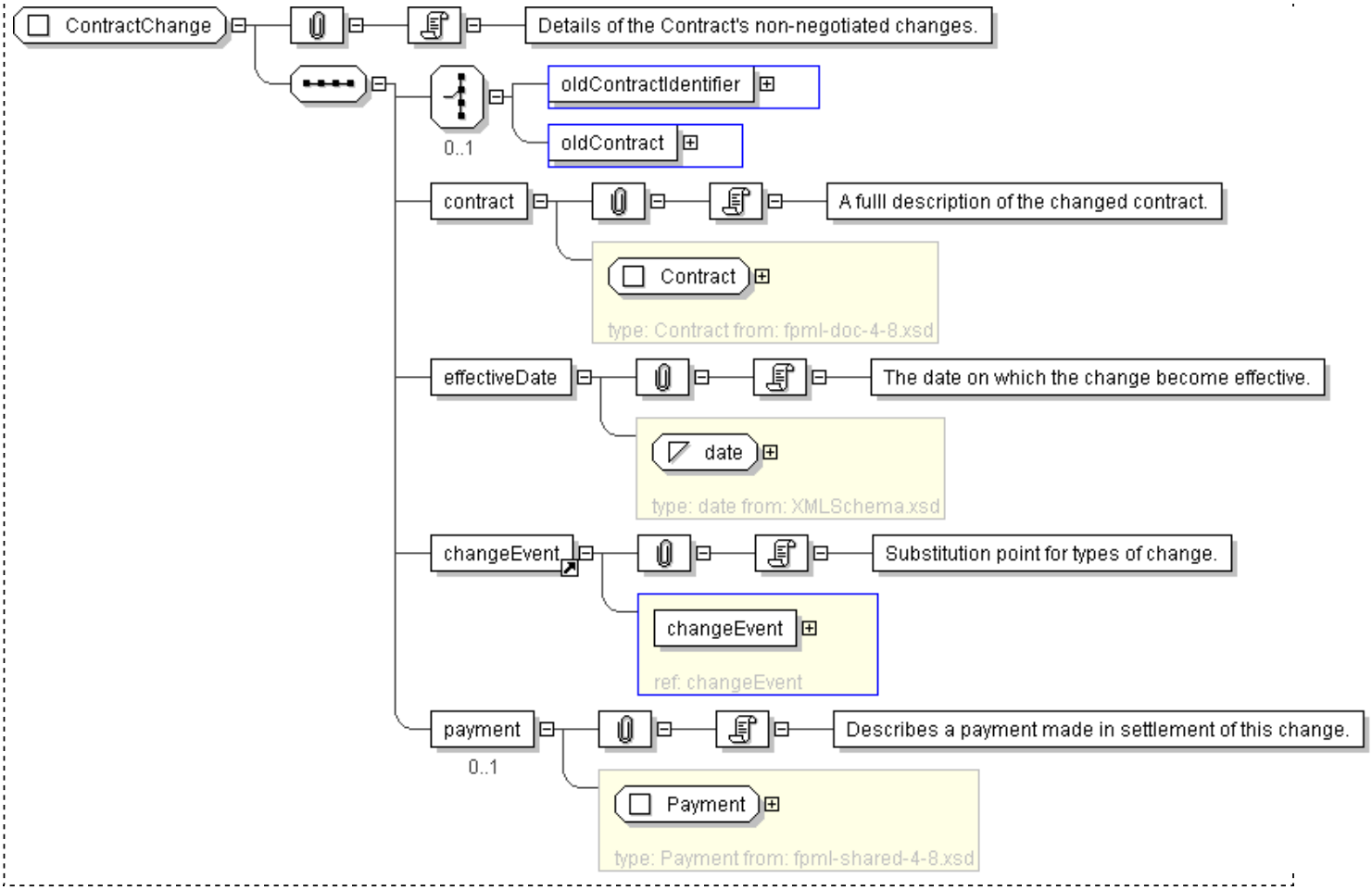
Super-types:	None
Sub-types:	None

Name	ContractChange
Used by (from the same schema document)	Complex Type ContractChanged , Complex Type ContractChangedCancelled
Abstract	no
Documentation	Details of the Contract's non-negotiated changes.

XML Instance Representation

```
<...>
   Start Choice [0..1]
      <oldContractIdentifier> PartyTradeIdentifier </oldContractIdentifier> [1]
      <oldContract> Contract </oldContract> [1]
   End Choice
   <contract> Contract </contract> [1]
   'A fulll description of the changed contract.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the change become effective.'

   <changeEvent> ... </changeEvent> [1]
   'Substitution point for types of change.'

   <payment> Payment </payment> [0..1]
   'Describes a payment made in settlement of this change.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractChange">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="oldContractIdentifier" type=" PartyTradeIdentifier "/>
      <xsd:element name="oldContract" type=" Contract "/>
    </xsd:choice>
    <xsd:element name="contract" type=" Contract "/>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
    <xsd:element ref=" changeEvent "/>
    <xsd:element name="payment" type=" Payment " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractChanged

[Table of contents]

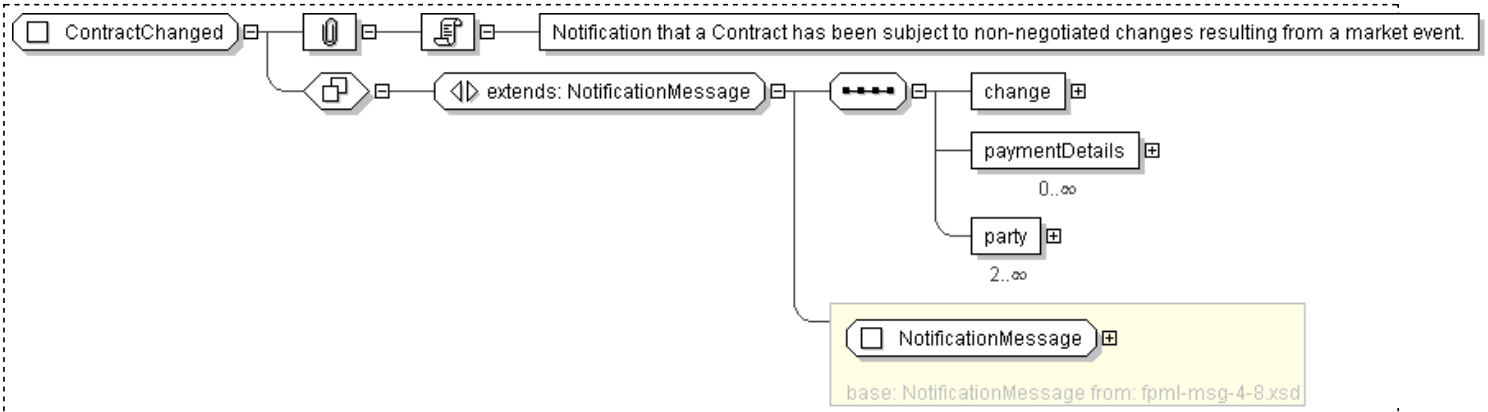
Super-types:	NotificationMessage < ContractChanged (by extension)
Sub-types:	None

Name	ContractChanged
Abstract	no
Documentation	Notification that a Contract has been subject to non-negotiated changes resulting from a market event.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <change> ContractChange </change> [1]
   'Details of the Contract\'s non-negotiated changes.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractChanged">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="change" type=" ContractChange "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractChangedCancelled

[Table of contents]

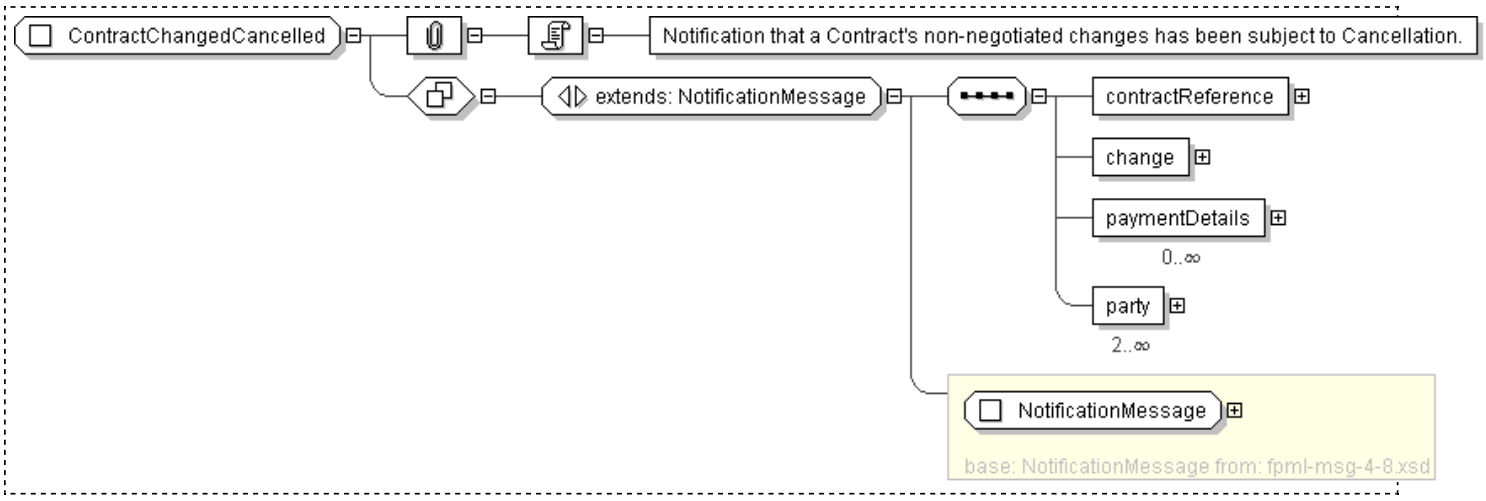
Super-types:	NotificationMessage < ContractChangedCancelled (by extension)
Sub-types:	None

Name	ContractChangedCancelled
Abstract	no
Documentation	Notification that a Contract's non-negotiated changes has been subject to Cancellation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'})
[1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the
 FpML schema and indicates the build number of the schema file. Every time FpML publishes a change
 to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build
 number is incremented. If no changes have been made between releases within a version (i.e. from
 Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Identification of a single Contract which is the subject of this message. Each Party may
   provide one to many identifiers for this contract.'

   <change> ContractChange </change> [1]
   'Details of the changes being cancelled.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractChangedCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="change" type=" ContractChange "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractCreated

[Table of contents]

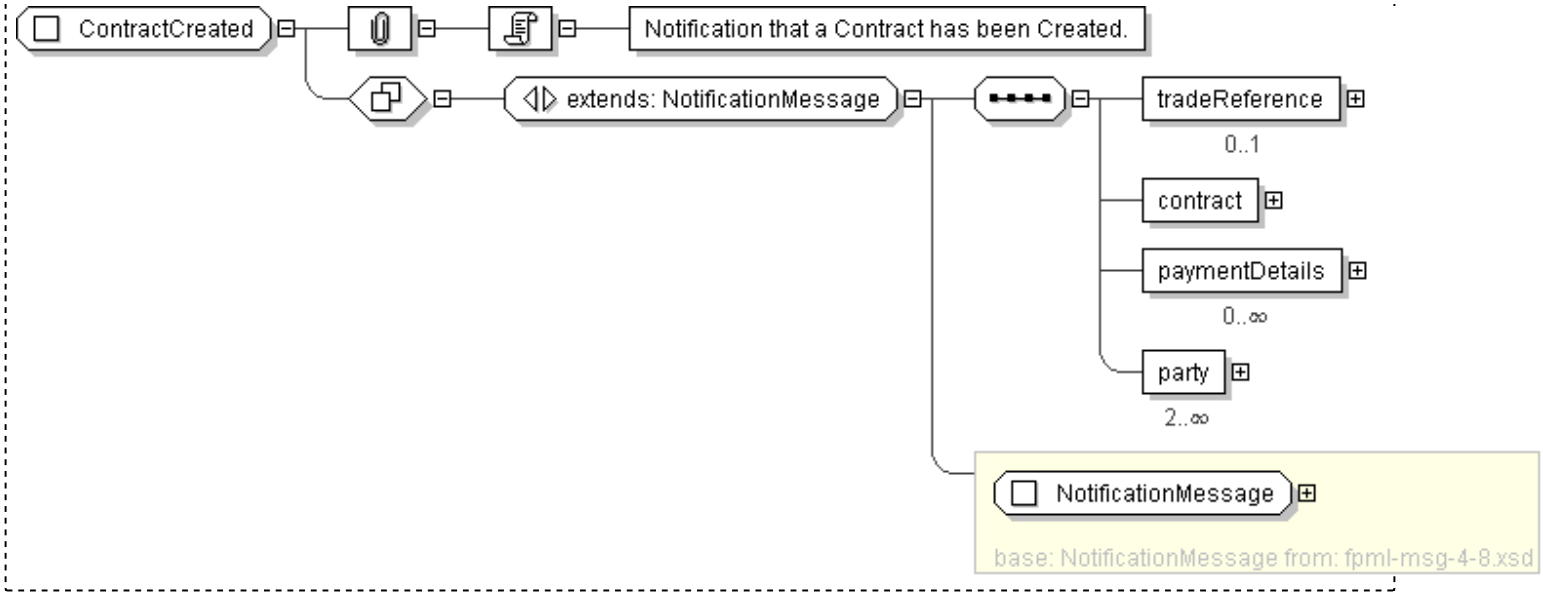
Super-types:	NotificationMessage < ContractCreated (by extension)
Sub-types:	None

Name	ContractCreated
Abstract	no
Documentation	Notification that a Contract has been Created.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeReference> PartyTradeIdentifiers </tradeReference> [0..1]
   'Optional reference to trade execution.'

   <contract> Contract </contract> [1]
   'Contract which has been created.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractCreated">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:element name="tradeReference" type=" PartyTradeIdentifiers "
            minOccurs="0"/>
            <xsd:element name="contract" type=" Contract "/>
            <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
            maxOccurs="unbounded"/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractFullTermination

[Table of contents]

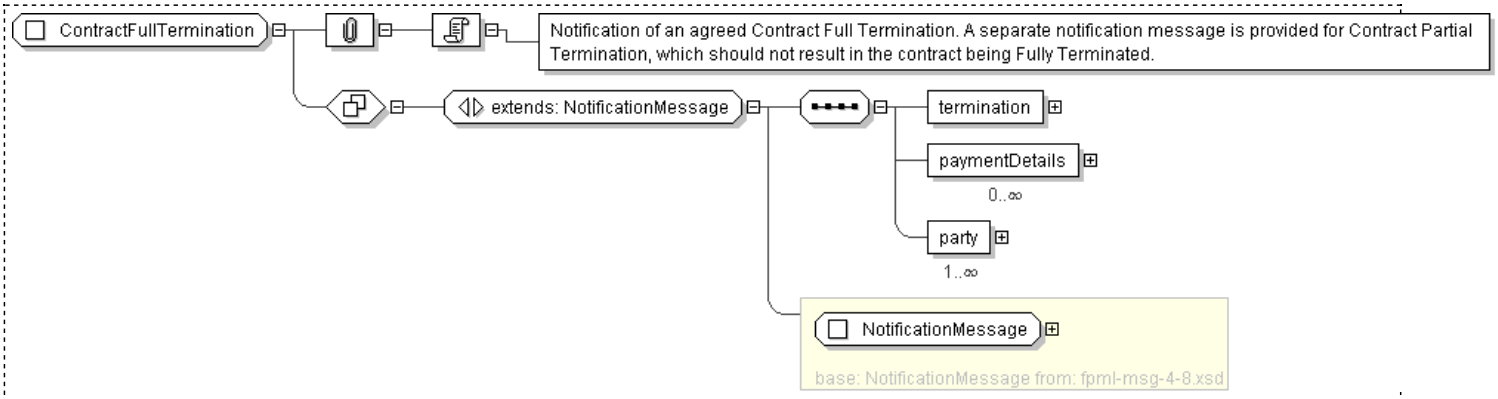
Super-types:	NotificationMessage < ContractFullTermination (by extension)
Sub-types:	None

Name	ContractFullTermination
Abstract	no
Documentation	Notification of an agreed Contract Full Termination. A separate notification message is provided for Contract Partial Termination, which should not result in the contract being Fully Terminated.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made
 between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays
 the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> ContractTermination </termination> [1]
   'Details of the full termination.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractFullTermination">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="termination" type=" ContractTermination "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractFullTerminationCancelled

[Table of contents]

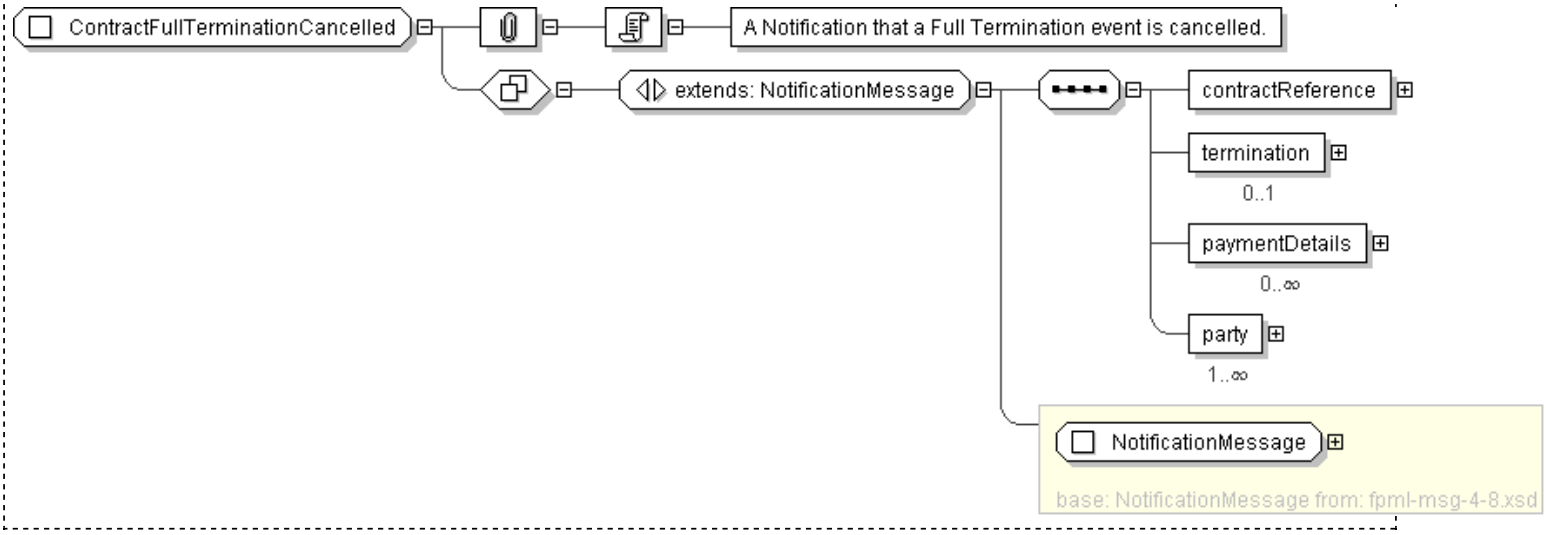
Super-types:	NotificationMessage < ContractFullTerminationCancelled (by extension)
Sub-types:	None

Name	ContractFullTerminationCancelled
Abstract	no
Documentation	A Notification that a Full Termination event is cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Reference to the contract affected by the cancellation.'

   <termination> ContractTermination </termination> [0..1]
   'Details of the Full Termination being cancelled.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractFullTerminationCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="termination" type=" ContractTermination " minOccurs="0"/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractIncreased

[Table of contents]

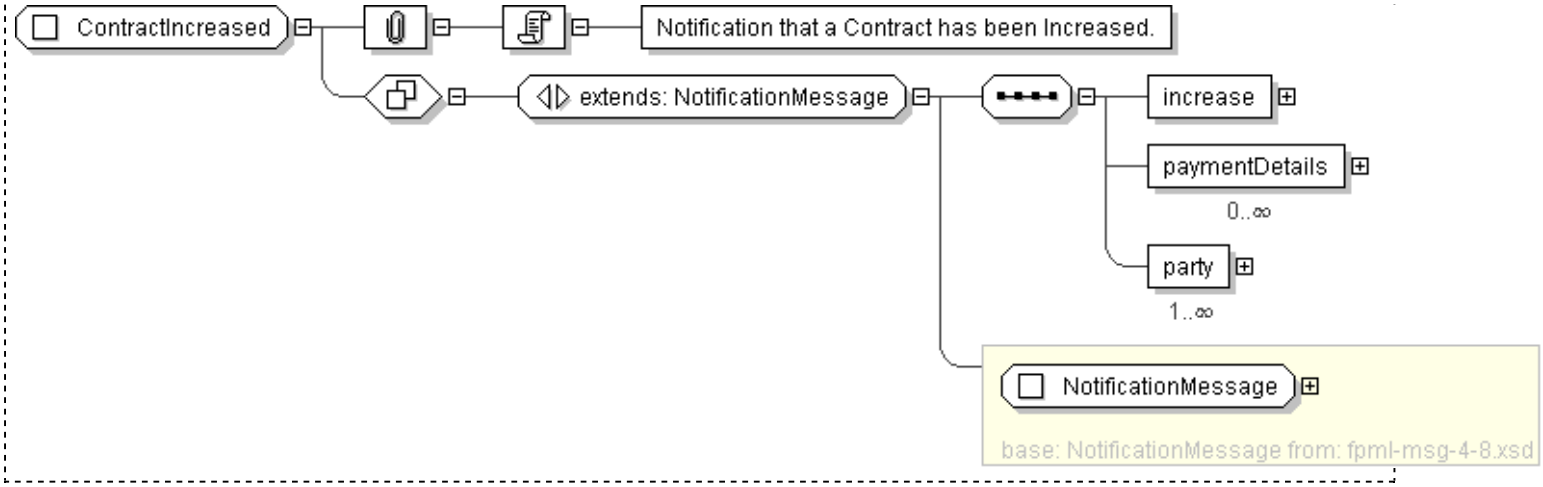
Super-types:	NotificationMessage < ContractIncreased (by extension)
Sub-types:	None

Name	ContractIncreased
Abstract	no
Documentation	Notification that a Contract has been Increased.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <increase> ChangeContractSize </increase> [1]
   'Increase Details.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractIncreased">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="increase" type=" ChangeContractSize "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractIncreasedCancelled

[Table of contents]

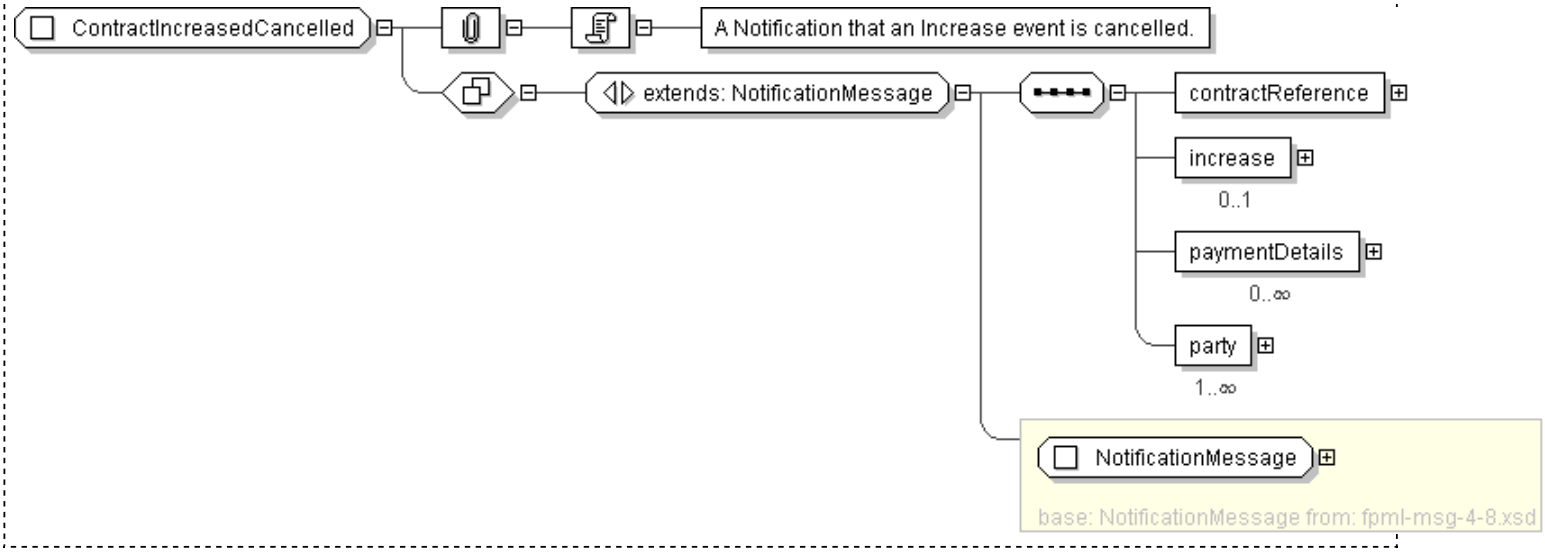
Super-types:	NotificationMessage < ContractIncreasedCancelled (by extension)
Sub-types:	None

Name	ContractIncreasedCancelled
Abstract	no
Documentation	A Notification that an Increase event is cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Reference to the contract affected by the cancellation.'

   <increase> ChangeContractSize </increase> [0..1]
   'Details of the Increase being cancelled.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractIncreasedCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="increase" type=" ChangeContractSize " minOccurs="0"/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractNovated

[Table of contents]

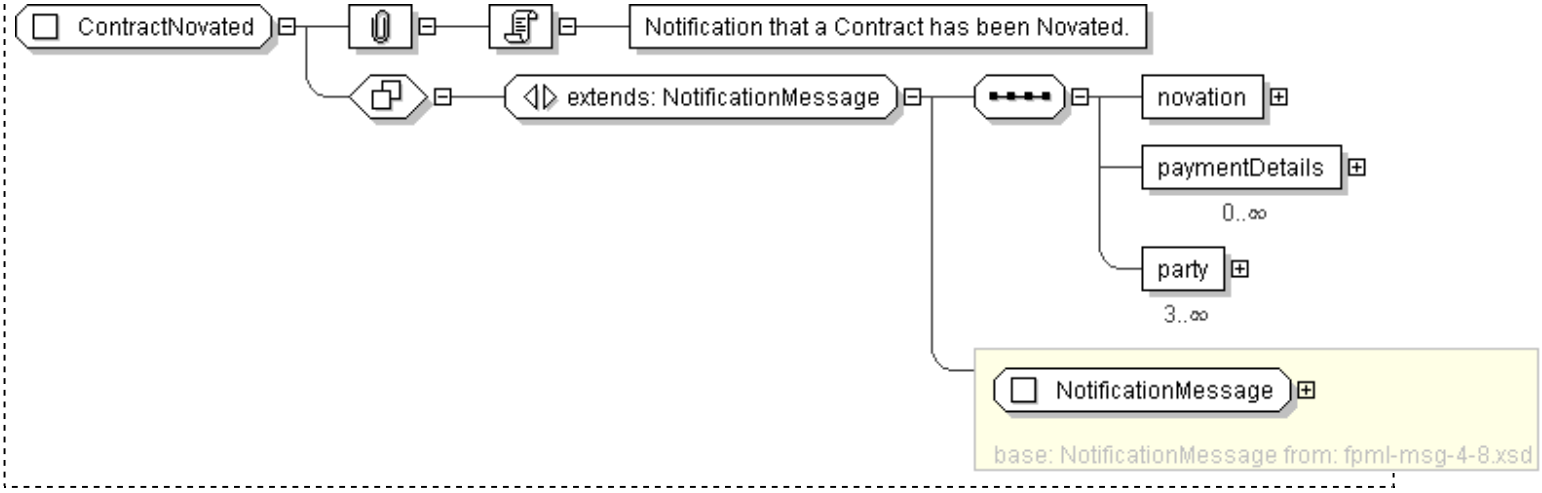
Super-types:	NotificationMessage < ContractNovated (by extension)
Sub-types:	None

Name	ContractNovated
Abstract	no
Documentation	Notification that a Contract has been Novated.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> ContractNovation </novation> [1]
   'Novation Details.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [3..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractNovated">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="novation" type=" ContractNovation "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " minOccurs="3" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractNovatedCancelled

[Table of contents]

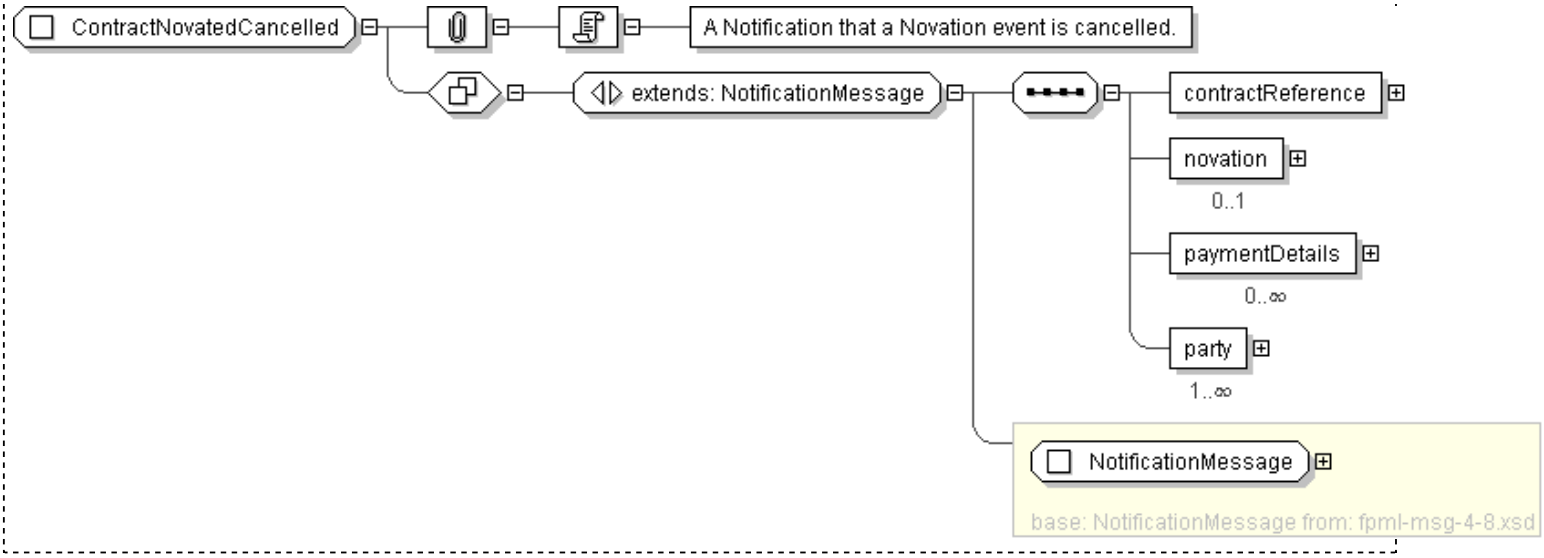
Super-types:	NotificationMessage < ContractNovatedCancelled (by extension)
Sub-types:	None

Name	ContractNovatedCancelled
Abstract	no
Documentation	A Notification that a Novation event is cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Reference to the contract affected by the cancellation.'

   <novation> ContractNovation </novation> [0..1]
   'Details of the Novation being cancelled.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractNovatedCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="novation" type=" ContractNovation " minOccurs="0"/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractPartialTermination

[Table of contents]

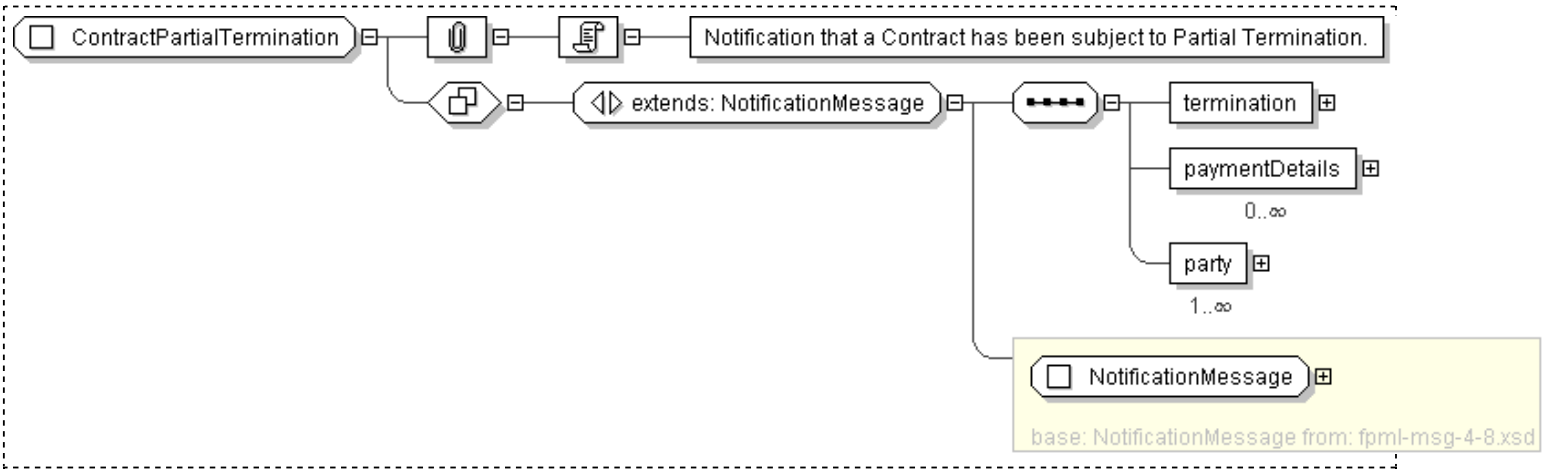
Super-types:	NotificationMessage < ContractPartialTermination (by extension)
Sub-types:	None

Name	ContractPartialTermination
Abstract	no
Documentation	Notification that a Contract has been subject to Partial Termination.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> ChangeContractSize </termination> [1]
   'Termination details.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractPartialTermination">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="termination" type=" ChangeContractSize "/>
        <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractPartialTerminationCancelled

[Table of contents]

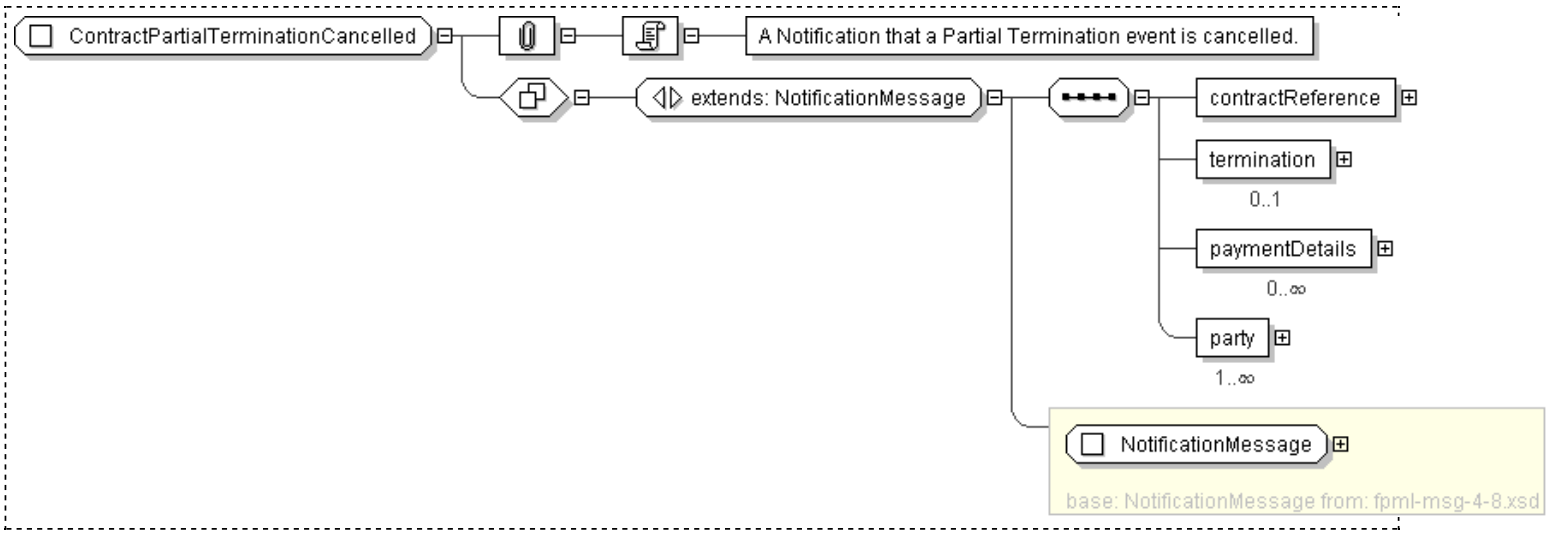
Super-types:	NotificationMessage < ContractPartialTerminationCancelled (by extension)
Sub-types:	None

Name	ContractPartialTerminationCancelled
Abstract	no
Documentation	A Notification that a Partial Termination event is cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'})
[1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify
 which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the
 FpML schema and indicates the build number of the schema file. Every time FpML publishes a change
 to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual
 build number is incremented. If no changes have been made between releases within a version (i.e.
 from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Reference to the contract affected by the cancellation.'

   <termination> ChangeContractSize </termination> [0..1]
   'Details of the Termination being cancelled.'

   <paymentDetails> PaymentDetails </paymentDetails> [0..*]
   'Details of the payments, like amount breakdowns, settlement information.'

   <party> Party </party> [1..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractPartialTerminationCancelled">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:element name="contractReference" type=" ContractReference "/>
            <xsd:element name="termination" type=" ChangeContractSize " minOccurs="0"/>
            <xsd:element name="paymentDetails" type=" PaymentDetails " minOccurs="0"
            maxOccurs="unbounded"/>
            <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractReferenceMessage

[Table of contents]

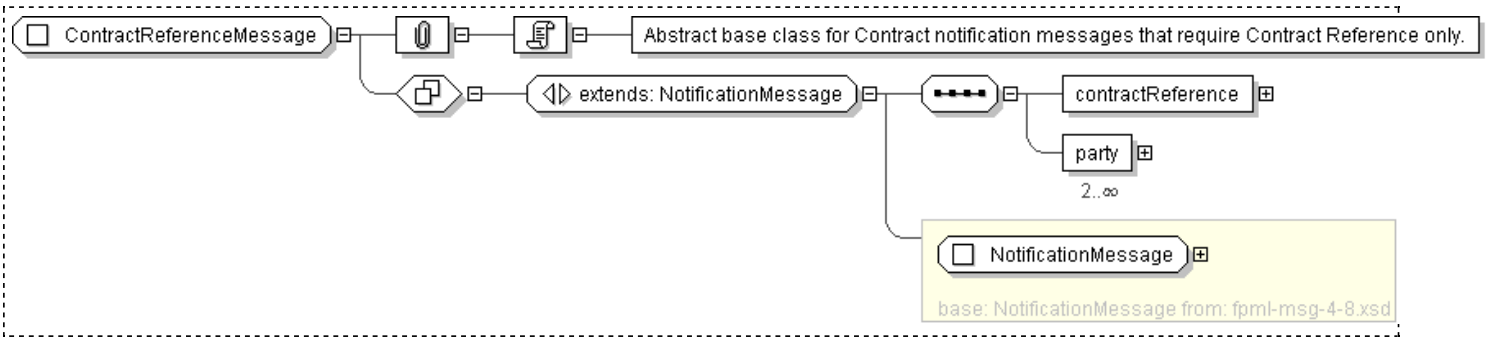
Super-types:	NotificationMessage < ContractReferenceMessage (by extension)
Sub-types:	• ContractCancelled (by extension)

Name	ContractReferenceMessage
Abstract	yes
Documentation	Abstract base class for Contract notification messages that require Contract Reference only.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <contractReference> ContractReference </contractReference> [1]
   'Identification of a single Contract which is the subject of this message. Each Party may provide
   one to many identifiers for this contract.'

   <party> Party </party> [2..*]
   'Identification of the Parties to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractReferenceMessage" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="contractReference" type=" ContractReference "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexChange

[Table of contents]

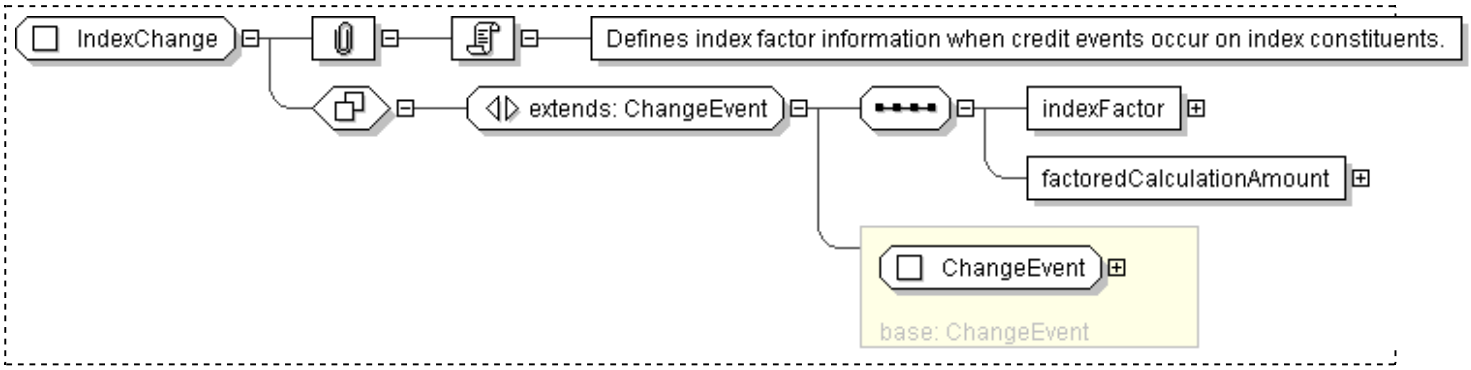
Super-types:	ChangeEvent < IndexChange (by extension)
Sub-types:	None

Name	IndexChange
Used by (from the same schema document)	Element indexChange
Abstract	no
Documentation	Defines index factor information when credit events occur on index constituents.

XML Instance Representation

```
<...>
   <indexFactor> RestrictedPercentage </indexFactor> [1]
   'The index version factor, expressed as a decimal, that multiplied by the original
   notional amount yields the notional amount covered by the seller. A factor of 75%
   would be represented as 0.75.'

   <factoredCalculationAmount> Money </factoredCalculationAmount> [1]
   'This relates only to CDS index contracts and represents the factored notional
   amount, obtained by multiplying the index factor to the contract notional amount.
   See indexFactor description above.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexChange">
   <xsd:complexContent>
      <xsd:extension base=" ChangeEvent ">
         <xsd:sequence>
            <xsd:element name="indexFactor" type=" RestrictedPercentage "/>
            <xsd:element name="factoredCalculationAmount" type=" Money "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: correlationSwap
- Global Definitions
  - Complex Type: CorrelationAmount
  - Complex Type: CorrelationLeg
  - Complex Type: CorrelationSwap
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 2527 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-eq-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 2527 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-eq-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema

component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

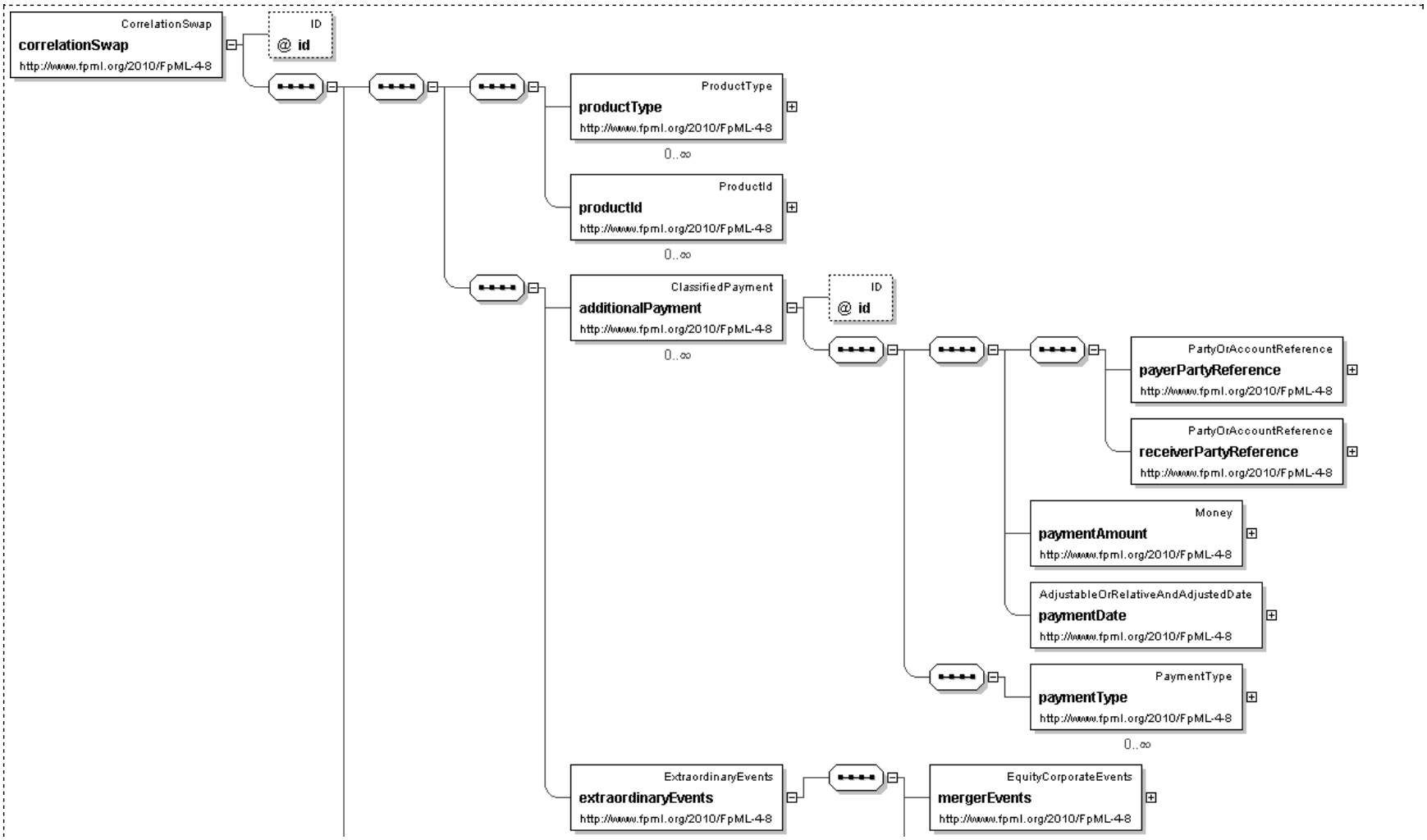
Element: correlationSwap

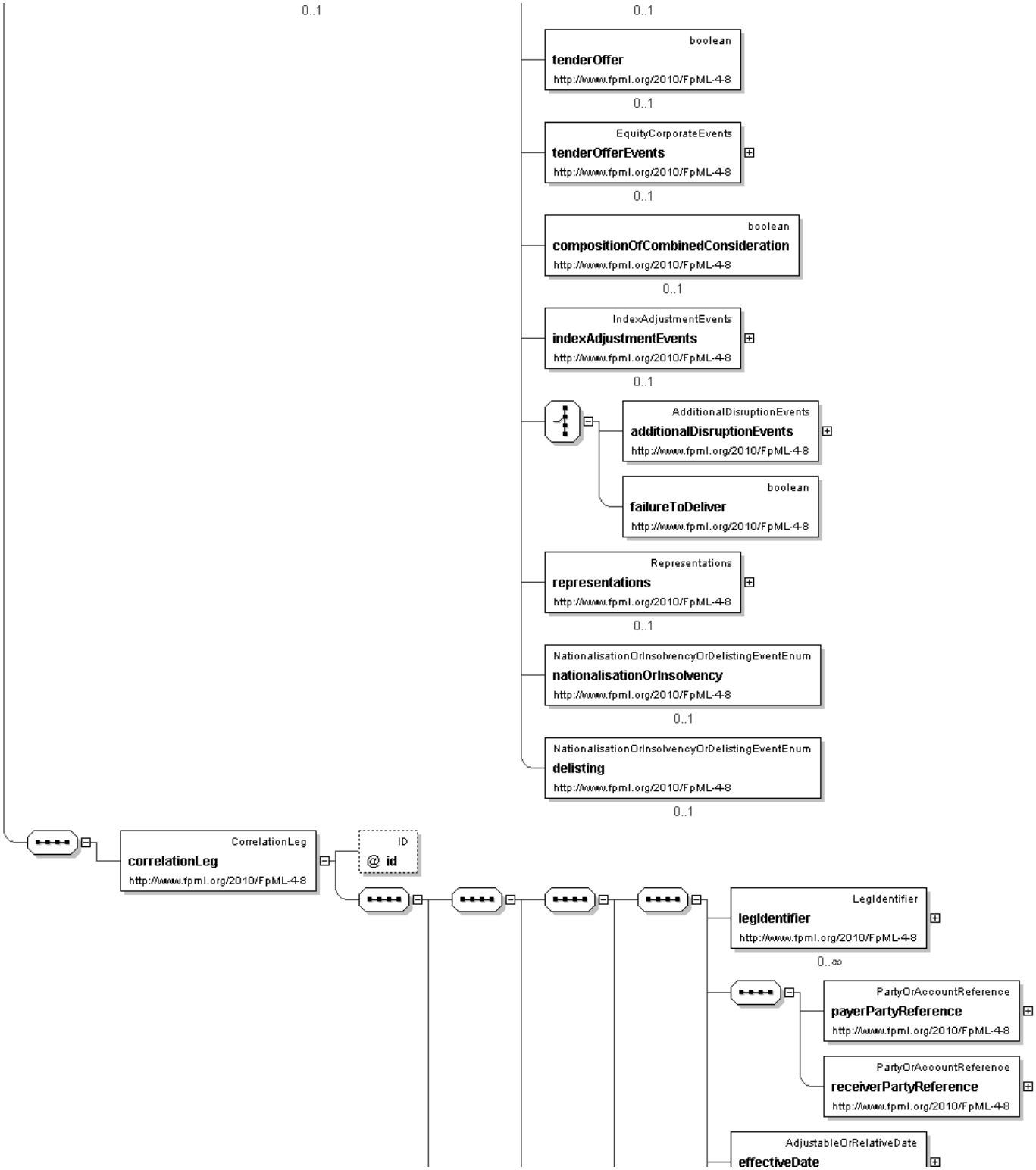
[Table of contents]

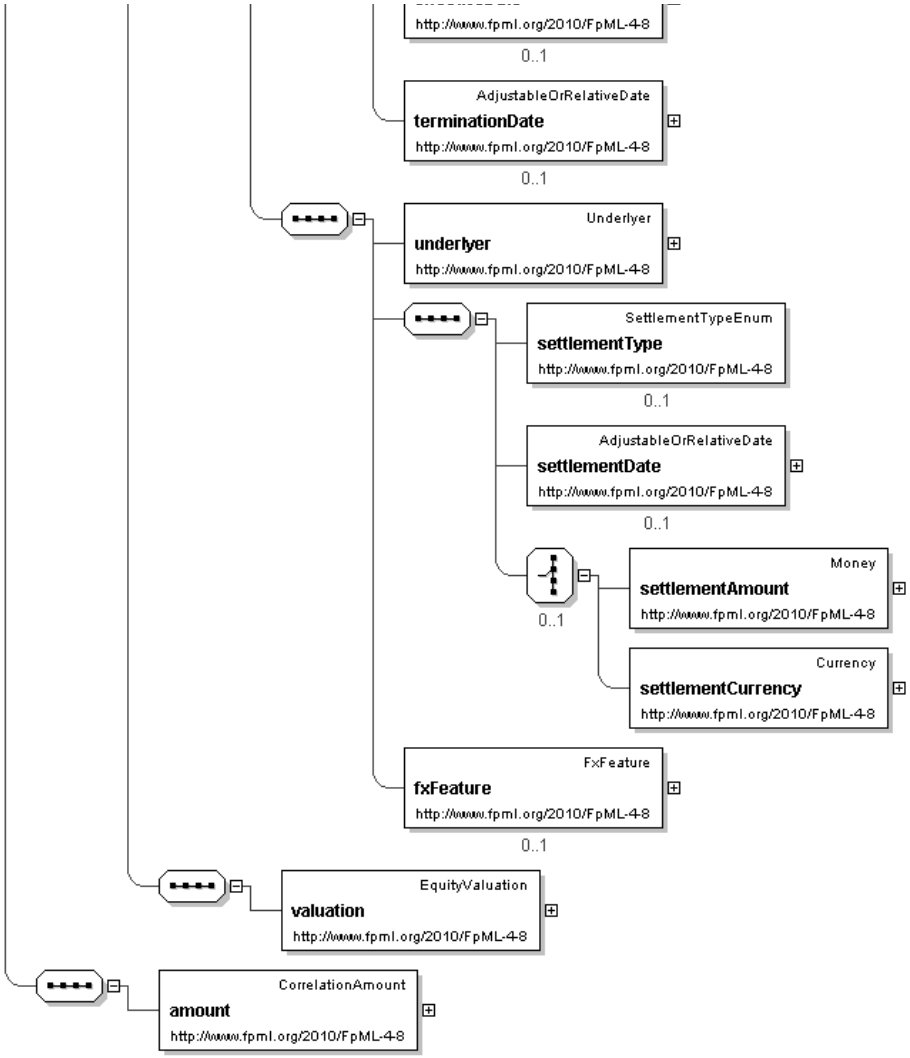
- *This element can be used wherever the following element is referenced:*
  - product

Name	correlationSwap
Type	CorrelationSwap
Nillable	no
Abstract	no
Documentation	Specifies the structure of a correlation swap.

Logical Diagram



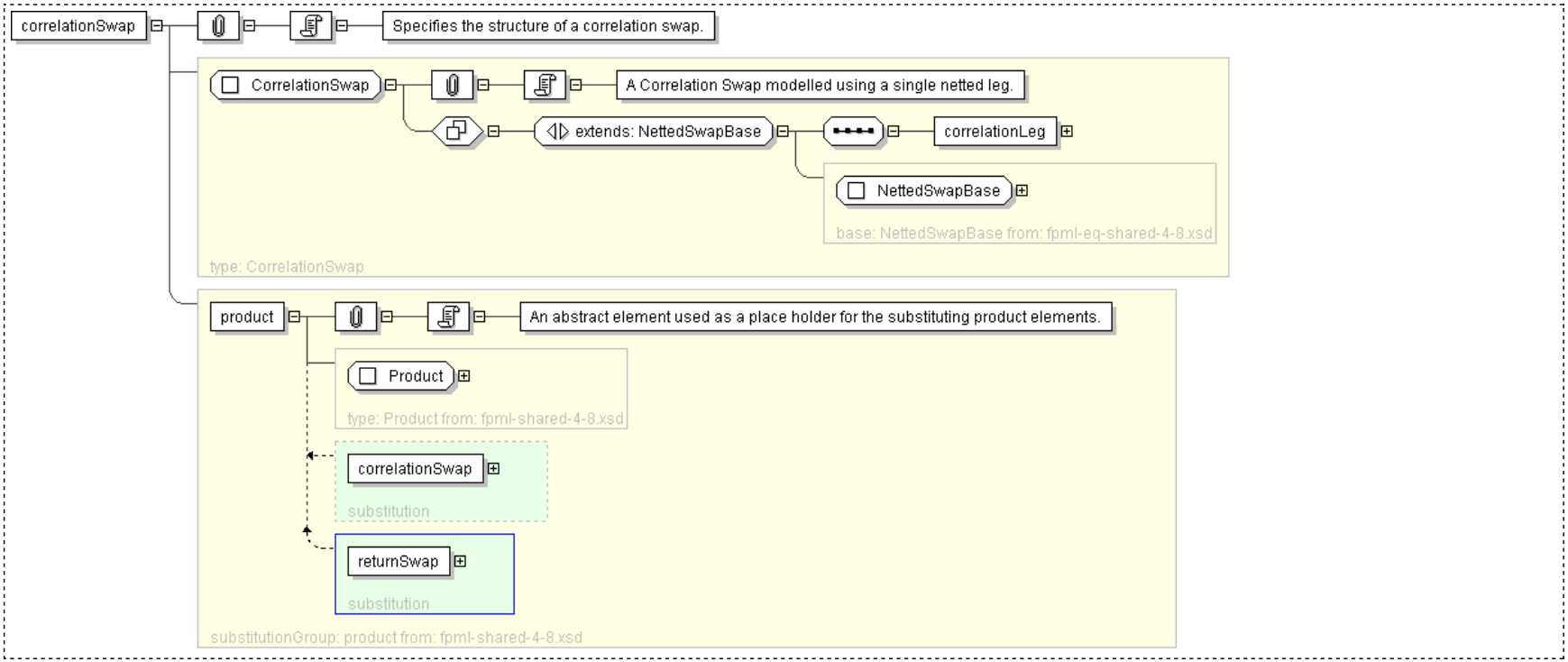

0..1
0..1
boolean
tenderOffer
http://www.fpml.org/2010/FpML-4-8
0..1
EquityCorporateEvents
tenderOfferEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
compositionOfCombinedConsideration
http://www.fpml.org/2010/FpML-4-8
0..1
IndexAdjustmentEvents
indexAdjustmentEvents
http://www.fpml.org/2010/FpML-4-8
0..1
AdditionalDisruptionEvents
additionalDisruptionEvents
http://www.fpml.org/2010/FpML-4-8
boolean
failureToDeliver
http://www.fpml.org/2010/FpML-4-8
Representations
representations
http://www.fpml.org/2010/FpML-4-8
0..1
NationalisationOrInsolvencyOrDelistingEventEnum
nationalisationOrInsolvency
http://www.fpml.org/2010/FpML-4-8
0..1
NationalisationOrInsolvencyOrDelistingEventEnum
delisting
http://www.fpml.org/2010/FpML-4-8
0..1
CorrelationLeg
correlationLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
LegIdentifier
legIdentifier
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
effectiveDate



XML Instance Representation

```
<correlationSwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain
    values for this element are not strictly an enumerated list.'

   <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
    'Specifies additional payment(s) between the principal parties to the netted swap.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
    'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be
    adjusted.'

   <correlationLeg> CorrelationLeg </correlationLeg> [1]
    'Correlation Leg. Correlation Buyer is deemed to be the Equity Amount Receiver, Correlation Seller is deemed to be the Equity Amount Payer.'

</correlationSwap>
```

Diagram



Schema Component Representation

```
<xsd:element name="correlationSwap" type=" CorrelationSwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CorrelationAmount

[Table of contents]

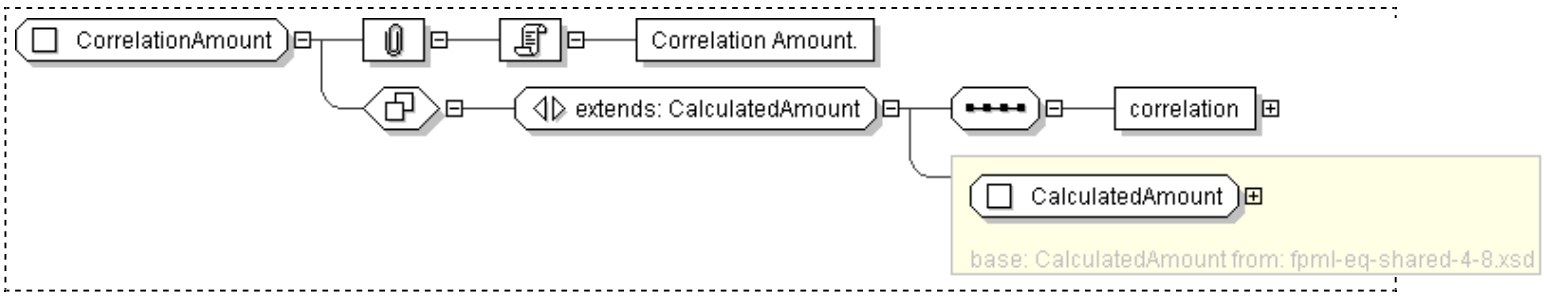
Super-types:	CalculatedAmount < CorrelationAmount (by extension)
Sub-types:	None

Name	CorrelationAmount
Used by (from the same schema document)	Complex Type CorrelationLeg
Abstract	no
Documentation	Correlation Amount.

XML Instance Representation

```
<...>
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for
   the purpose of calculating the amount.'

   <observationStartDate> AdjustableOrRelativeDate </observationStartDate> [0..1]
   'The start of the period over which observations are made which are used in the
   calculation Used when the observation start date differs from the trade date such as
   for forward starting swaps.'

   <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
   'If present and true, then options exchange dividends are applicable.'

   <additionalDividends> xsd:boolean </additionalDividends> [0..1]
   'If present and true, then additional dividends are applicable.'

   <allDividends> xsd:boolean </allDividends> [0..1]
   'Represents the European Master Confirmation value of \'All Dividends\' which, when
   applicable, signifies that, for a given Ex-Date, the daily observed Share Price for
   that day is adjusted (reduced) by the cash dividend and/or the cash value of any non
   cash dividend per Share (including Extraordinary Dividends) declared by the Issuer.'

   <correlation> Correlation </correlation> [1]
   'Specifies Correlation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CorrelationAmount">
   <xsd:complexContent>
      <xsd:extension base=" CalculatedAmount ">
         <xsd:sequence>
            <xsd:element name="correlation" type=" Correlation "/>
         </xsd:sequence>
      </xsd:extension>
```

```
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CorrelationLeg

[Table of contents]

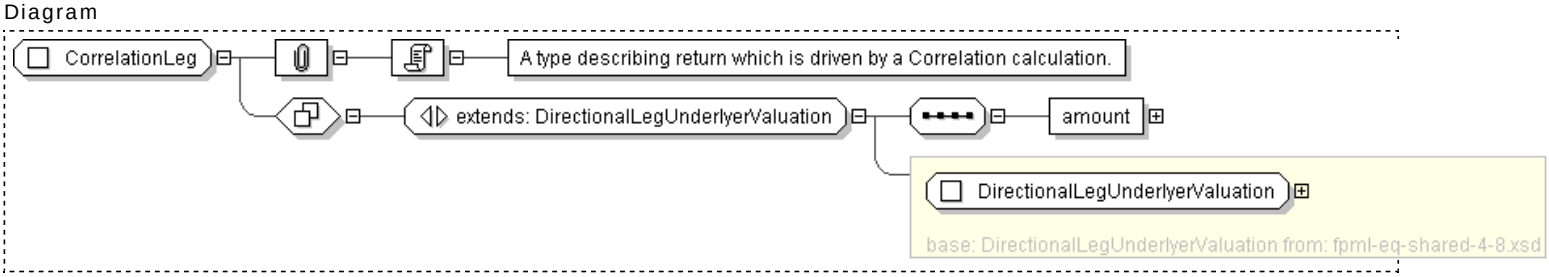
Super-types:	DirectionalLegUnderlyerValuation < CorrelationLeg (by extension)
Sub-types:	None

Name	CorrelationLeg
Used by (from the same schema document)	Complex Type CorrelationSwap
Abstract	no
Documentation	A type describing return which is driven by a Correlation calculation.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
    'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
    'Specifies the effective date of this leg of the swap. When defined in relation to a date specified
    somewhere else in the document (through the relativeDate component), this element will typically
    point to the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
    'Specifies the termination date of this leg of the swap. When defined in relation to a date
    specified somewhere else in the document (through the relativeDate component), this element will
    typically point to the termination date of the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlyer of the leg.'

   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
          'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
          'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <fxFeature> FxFeature </fxFeature> [0..1]
    'Quanto, Composite, or Cross Currency FX features.'

   <valuation> EquityValuation </valuation> [1]
    'Valuation of the underlyer.'

   <amount> CorrelationAmount </amount> [1]
    'Specifies, in relation to each Equity Payment Date, the Equity Amount to which the Equity Payment
    Date relates. Unless otherwise specified, this term has the meaning defined in the ISDA 2002 Equity
    Derivatives Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CorrelationLeg">
  <xsd:complexContent>
    <xsd:extension base=" DirectionalLegUnderlyerValuation ">
      <xsd:sequence>
        <xsd:element name="amount" type=" CorrelationAmount "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CorrelationSwap

[Table of contents]

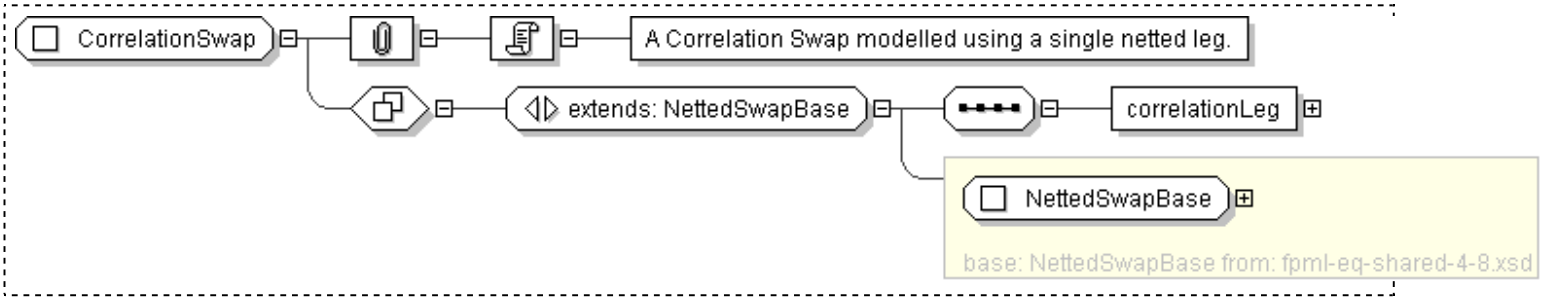
Super-types:	NettedSwapBase < CorrelationSwap (by extension)
Sub-types:	None

Name	CorrelationSwap
Used by (from the same schema document)	Element correlationSwap
Abstract	no
Documentation	A Correlation Swap modelled using a single netted leg.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for this
   element are not strictly an enumerated list.'

   <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
   'Specifies additional payment(s) between the principal parties to the netted
   swap.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
   'Where the underlying is shares, specifies events affecting the issuer of those
   shares that may require the terms of the transaction to be adjusted.'

   <correlationLeg> CorrelationLeg </correlationLeg> [1]
   'Correlation Leg. Correlation Buyer is deemed to be the Equity Amount Receiver,
   Correlation Seller is deemed to be the Equity Amount Payer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CorrelationSwap">
   <xsd:complexContent>
      <xsd:extension base=" NettedSwapBase ">
         <xsd:sequence>
            <xsd:element name="correlationLeg" type=" CorrelationLeg "/>
         </xsd:sequence>
      </xsd:extension>
```

```
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: bankruptcy
  - Element: creditEvent
  - Element: creditEventNotice
  - Element: failureToPay
  - Element: obligationAcceleration
  - Element: obligationDefault
  - Element: repudiationMoratorium
  - Element: restructuring
- Global Definitions
  - Complex Type: BankruptcyEvent
  - Complex Type: CreditEvent
  - Complex Type: CreditEventNoticeDocument
  - Complex Type: CreditEventNotification
  - Complex Type: FailureToPayEvent
  - Complex Type: Language
  - Complex Type: ObligationAccelerationEvent
  - Complex Type: ObligationDefaultEvent
  - Complex Type: RepudiationMoratoriumEvent
  - Complex Type: Resource
  - Complex Type: ResourceId
  - Complex Type: ResourceLength
  - Complex Type: RestructuringEvent
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore

fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given

type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

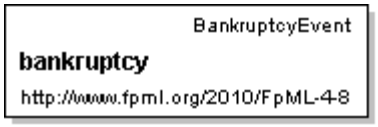
Element: bankruptcy

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

Name	bankruptcy
Type	BankruptcyEvent
Nillable	no
Abstract	no

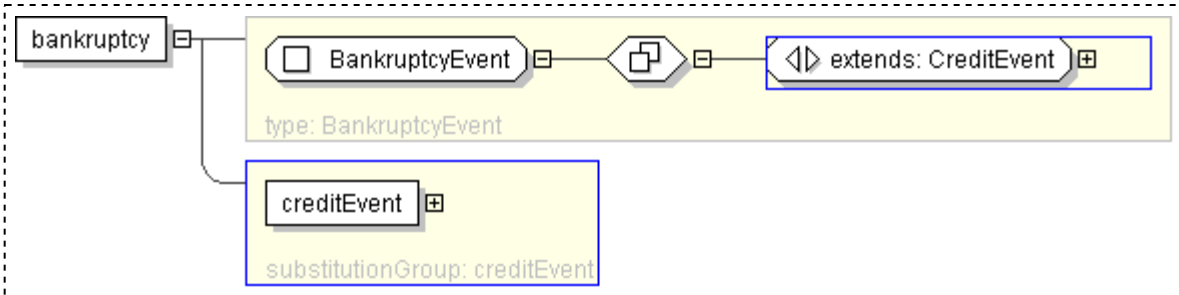
Logical Diagram



XML Instance Representation

```
<bankruptcy/>
```

Diagram



Schema Component Representation

```
<xsd:element name="bankruptcy" type=" BankruptcyEvent "
substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: creditEvent

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - bankruptcy
  - failureToPay
  - obligationAcceleration
  - obligationDefault
  - repudiationMoratorium
  - restructuring

Name	creditEvent
Used by (from the same schema document)	Complex Type CreditEventNoticeDocument
Type	CreditEvent
Nillable	no
Abstract	yes

Logical Diagram

XML Instance Representation

```
<creditEvent/>
```

Diagram

Schema Component Representation

```
<xsd:element name="creditEvent" type=" CreditEvent " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

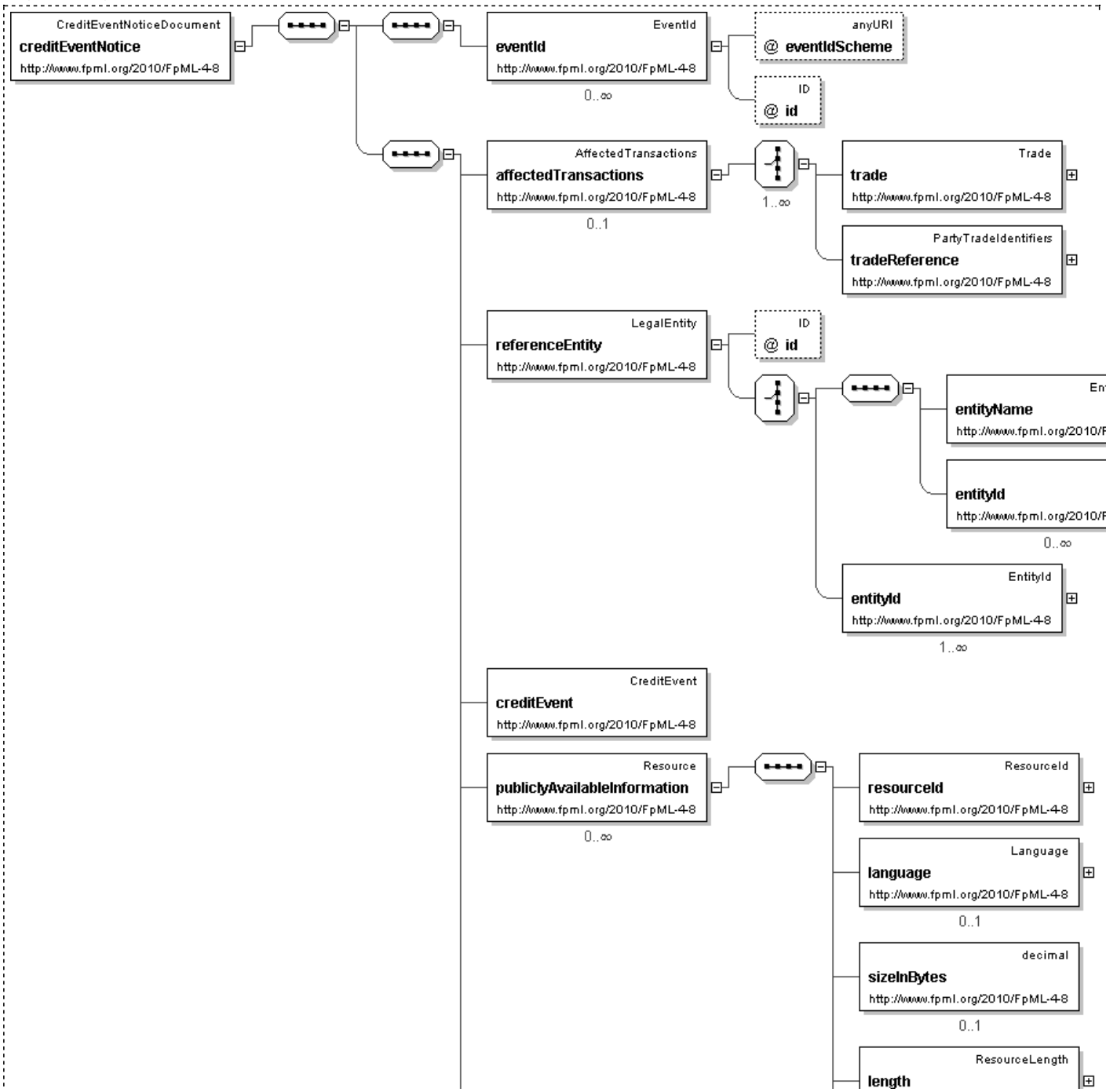
Element: creditEventNotice

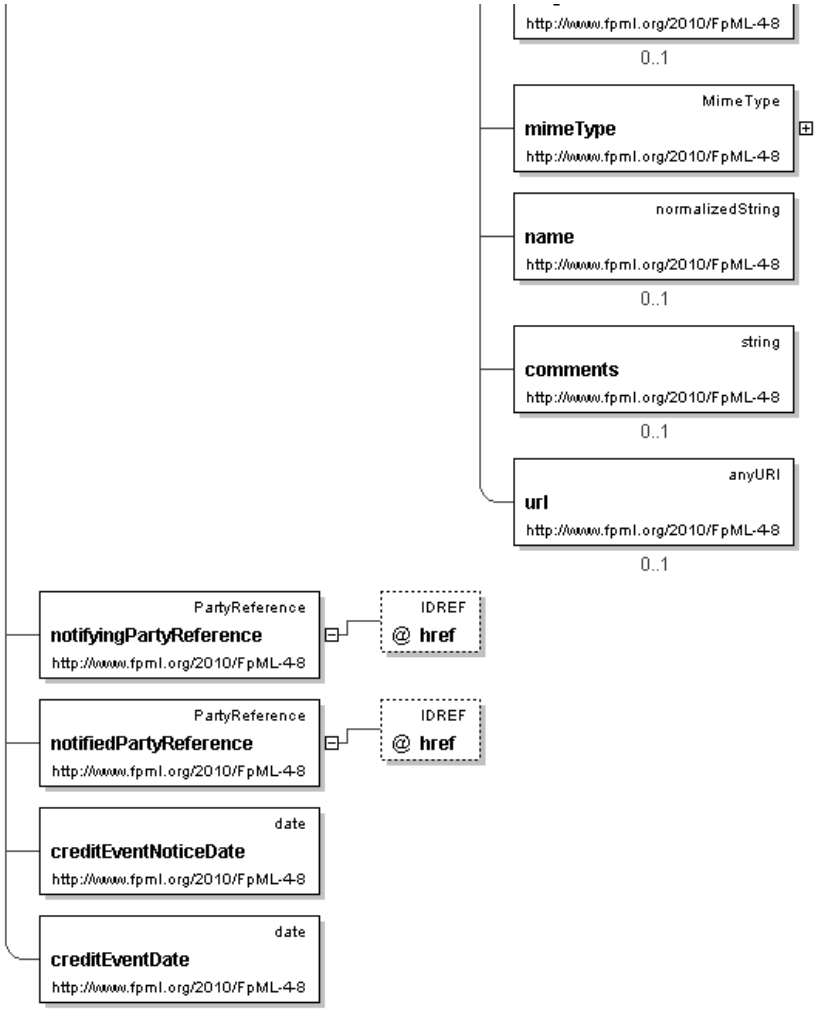
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - event

Name	creditEventNotice
Type	CreditEventNoticeDocument
Nillable	no
Abstract	no
Documentation	A global element used to hold CENs.

Logical Diagram

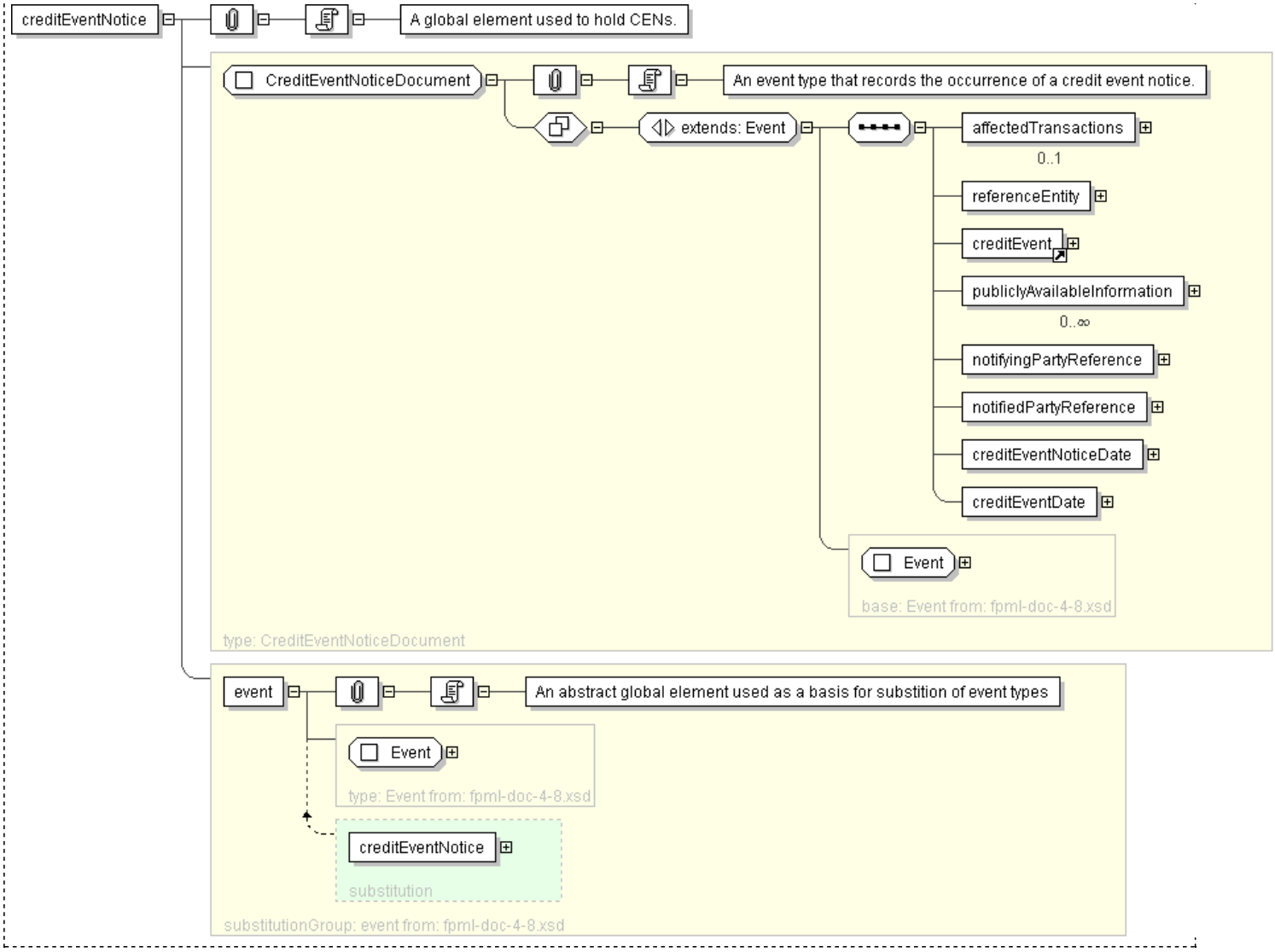




XML Instance Representation

```
<creditEventNotice>
   <eventId> EventId </eventId> [0..*]
   ''

   <affectedTransactions> AffectedTransactions </affectedTransactions> [0..1]
   'Trades affected by this event.'

   <referenceEntity> LegalEntity </referenceEntity> [1]
   <creditEvent> ... </creditEvent> [1]
   <publiclyAvailableInformation> Resource </publiclyAvailableInformation> [0..*]
   'A public information source, e.g. a particular newspaper or electronic news service, that may publish
   relevant information used in the determination of whether or not a credit event has occurred.'

   <notifyingPartyReference> PartyReference </notifyingPartyReference> [1]
   <notifiedPartyReference> PartyReference </notifiedPartyReference> [1]
   <creditEventNoticeDate> xsd:date </creditEventNoticeDate> [1]
   <creditEventDate> xsd:date </creditEventDate> [1]
</creditEventNotice>
```

Diagram



Schema Component Representation

```
<xsd:element name="creditEventNotice" type=" CreditEventNoticeDocument " substitutionGroup="event"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

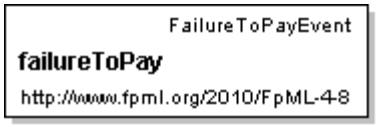
Element: failureToPay

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

Name	failureToPay
Type	FailureToPayEvent
Nillable	no
Abstract	no

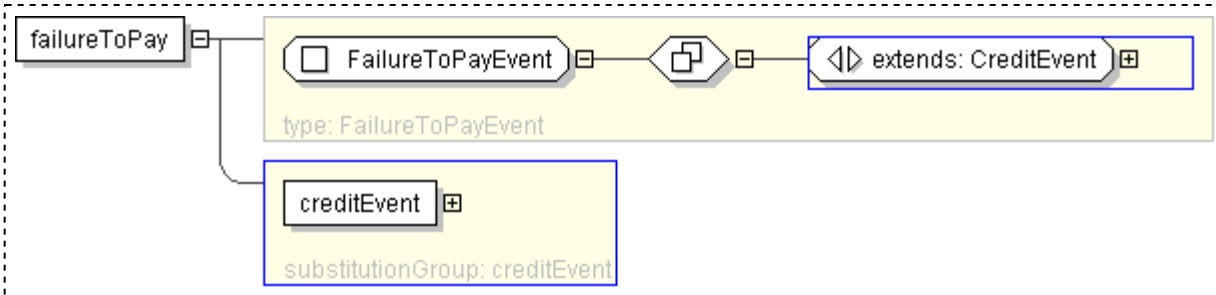
Logical Diagram



XML Instance Representation

```
<failureToPay/>
```

Diagram



Schema Component Representation

```
<xsd:element name="failureToPay" type=" FailureToPayEvent "
substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

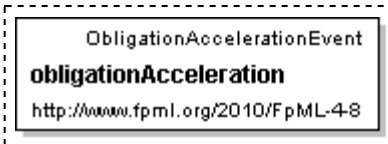
Element: obligationAcceleration

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

Name	obligationAcceleration
Type	ObligationAccelerationEvent
Nillable	no
Abstract	no

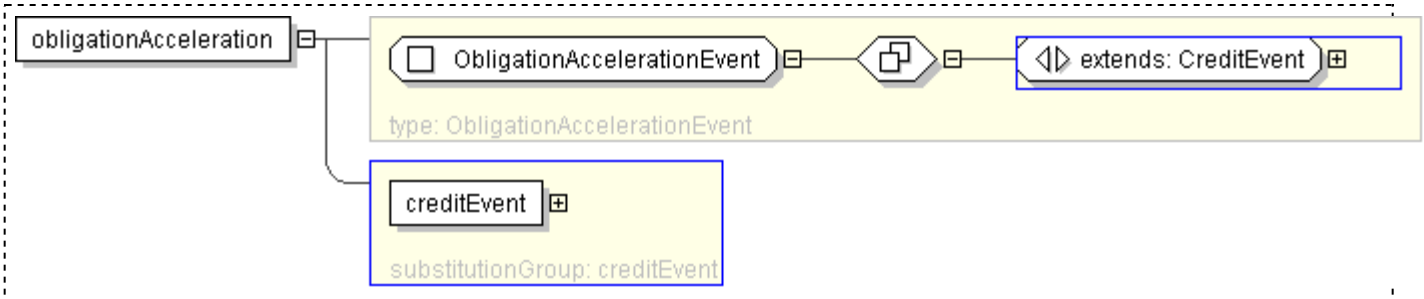
Logical Diagram



XML Instance Representation

```
<obligationAcceleration/>
```

Diagram



Schema Component Representation

```
<xsd:element name="obligationAcceleration" type=" ObligationAccelerationEvent "
substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: obligationDefault

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

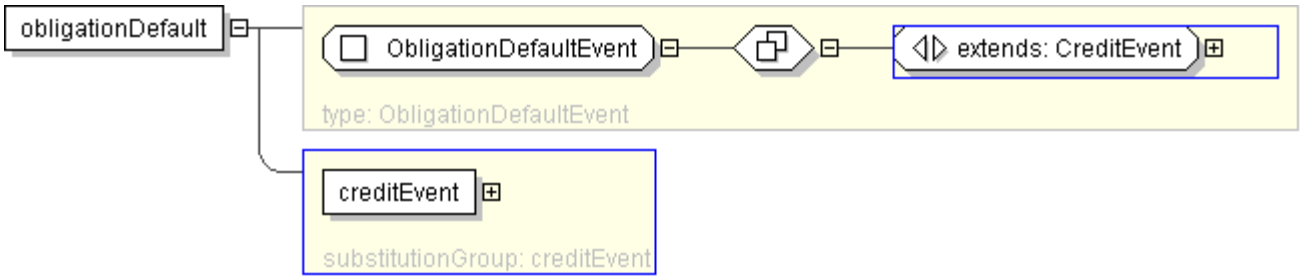
Name	obligationDefault
Type	ObligationDefaultEvent
Nillable	no
Abstract	no

Logical Diagram

XML Instance Representation

```
<obligationDefault/>
```

Diagram



Schema Component Representation

```
<xsd:element name="obligationDefault" type=" ObligationDefaultEvent "
substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: repudiationMoratorium

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

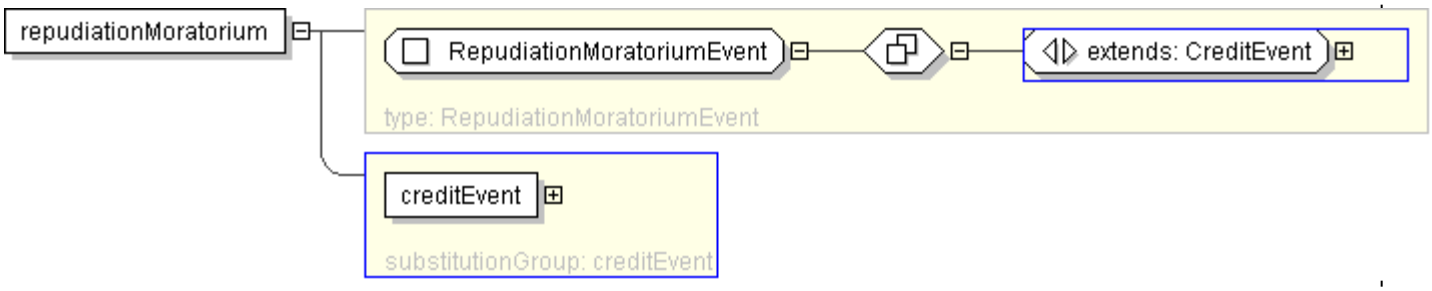
Name	repudiationMoratorium
Type	RepudiationMoratoriumEvent
Nillable	no
Abstract	no

Logical Diagram

XML Instance Representation

```
<repudiationMoratorium/>
```

Diagram



Schema Component Representation

```
<xsd:element name="repudiationMoratorium" type=" RepudiationMoratoriumEvent "
substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

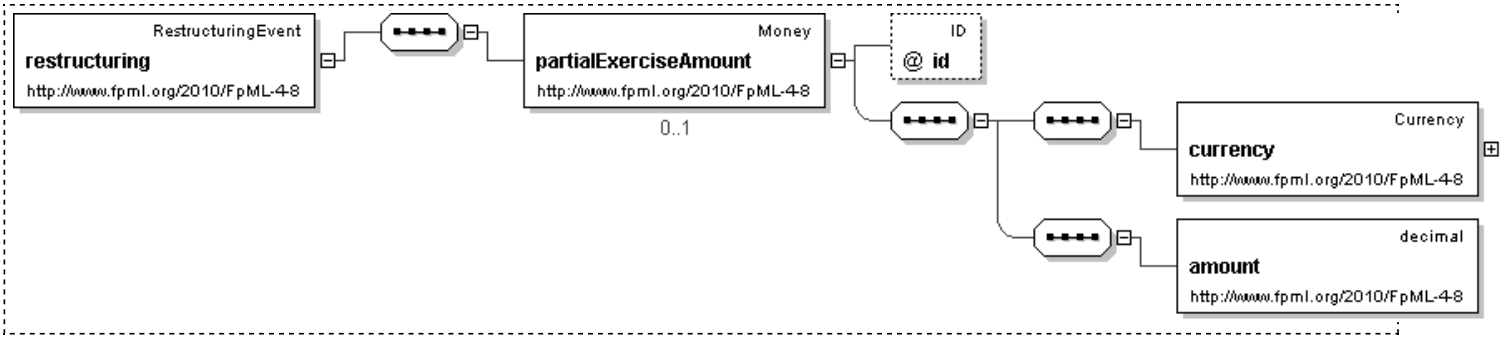
Element: restructuring

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - creditEvent

Name	restructuring
Type	RestructuringEvent
Nillable	no
Abstract	no

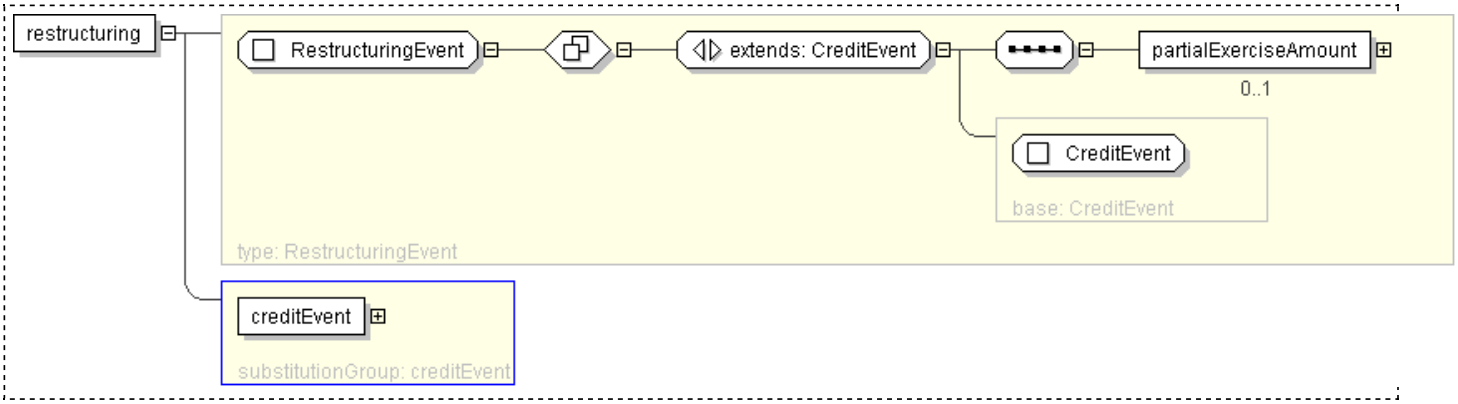
Logical Diagram



XML Instance Representation

```
<restructuring>
   <partialExerciseAmount> Money </partialExerciseAmount> [0..1]
</restructuring>
```

Diagram



Schema Component Representation

```
<xsd:element name="restructuring" type=" RestructuringEvent " substitutionGroup="creditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BankruptcyEvent

[Table of contents]

Super-types:	CreditEvent < BankruptcyEvent (by extension)
Sub-types:	None

Name	BankruptcyEvent
Used by (from the same schema document)	Element bankruptcy
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="BankruptcyEvent">
   <xsd:complexContent>
      <xsd:extension base=" CreditEvent "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEvent

[Table of contents]

Super-types: None

Sub-types:

- BankruptcyEvent (by extension)
- FailureToPayEvent (by extension)
- ObligationAccelerationEvent (by extension)
- ObligationDefaultEvent (by extension)
- RepudiationMoratoriumEvent (by extension)
- RestructuringEvent (by extension)

Name	CreditEvent
Used by (from the same schema document)	Element creditEvent
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="CreditEvent"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEventNoticeDocument

[Table of contents]

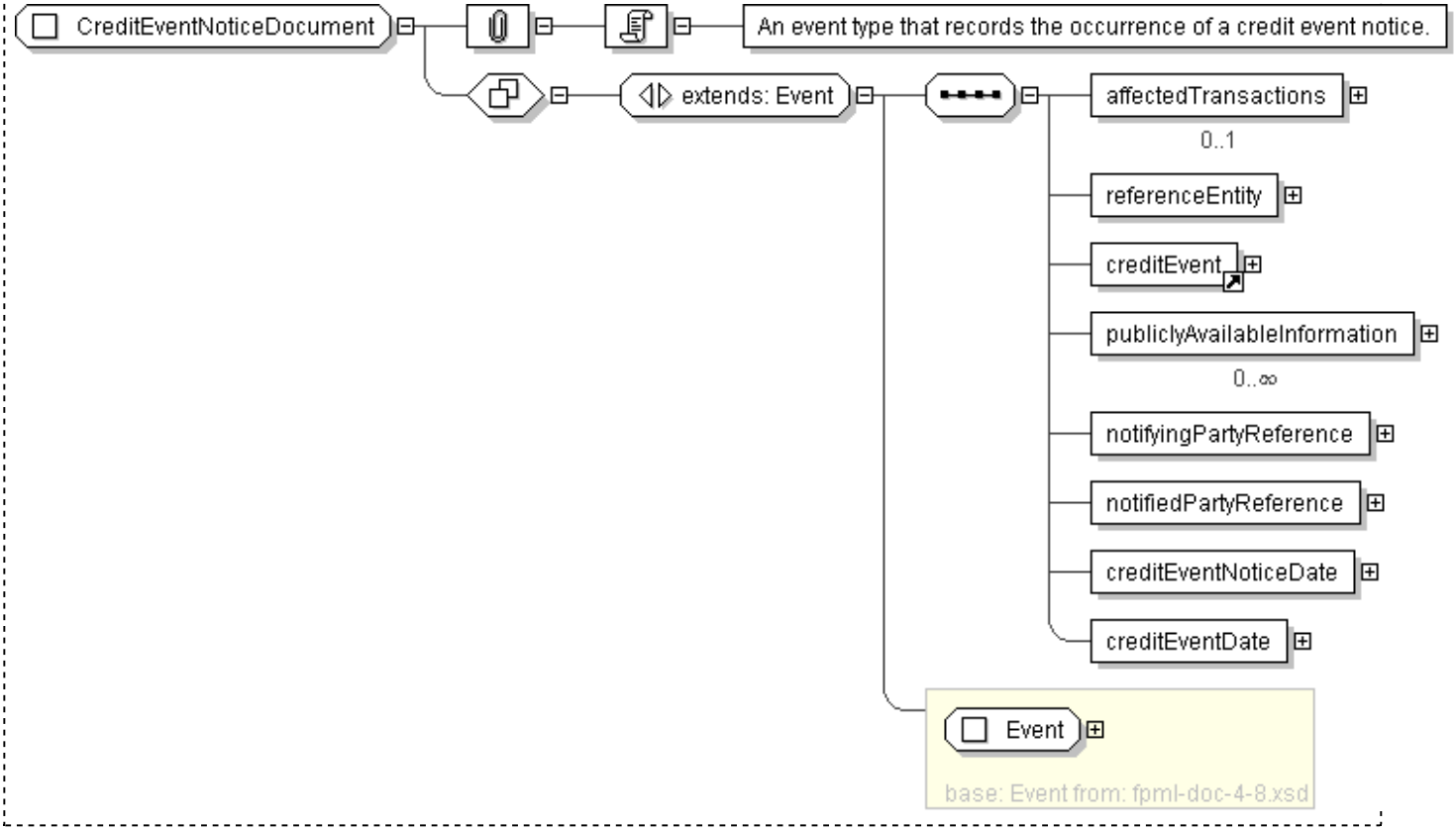
Super-types:	Event < CreditEventNoticeDocument (by extension)
Sub-types:	None

Name	CreditEventNoticeDocument
Used by (from the same schema document)	Complex Type CreditEventNotification , Element creditEventNotice
Abstract	no
Documentation	An event type that records the occurrence of a credit event notice.

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

   <affectedTransactions> AffectedTransactions </affectedTransactions> [0..1]
   'Trades affected by this event.'

   <referenceEntity> LegalEntity </referenceEntity> [1]
   <creditEvent> ... </creditEvent> [1]
   <publiclyAvailableInformation> Resource </publiclyAvailableInformation> [0..*]
   'A public information source, e.g. a particular newspaper or electronic news
   service, that may publish relevant information used in the determination of
   whether or not a credit event has occurred.'

   <notifyingPartyReference> PartyReference </notifyingPartyReference> [1]
   <notifiedPartyReference> PartyReference </notifiedPartyReference> [1]
   <creditEventNoticeDate> xsd:date </creditEventNoticeDate> [1]
   <creditEventDate> xsd:date </creditEventDate> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditEventNoticeDocument">
  <xsd:complexContent>
    <xsd:extension base=" Event ">
      <xsd:sequence>
        <xsd:element name="affectedTransactions" type=" AffectedTransactions "
        minOccurs="0"/>
        <xsd:element name="referenceEntity" type=" LegalEntity "/>
        <xsd:element ref=" creditEvent "/>
        <xsd:element name="publiclyAvailableInformation" type=" Resource "
        minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="notifyingPartyReference" type=" PartyReference "/>
        <xsd:element name="notifiedPartyReference" type=" PartyReference "/>
        <xsd:element name="creditEventNoticeDate" type=" xsd:date "/>
        <xsd:element name="creditEventDate" type=" xsd:date "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEventNotification

[Table of contents]

Super-types:	NotificationMessage < CreditEventNotification (by extension)
Sub-types:	None

Name	CreditEventNotification
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <creditEventNotice> CreditEventNoticeDocument </creditEventNotice> [1]
   <party> Party </party> [2..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="CreditEventNotification">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:element name="creditEventNotice" type=" CreditEventNoticeDocument "/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FailureToPayEvent

[Table of contents]

Super-types:	CreditEvent < FailureToPayEvent (by extension)
Sub-types:	None

Name	FailureToPayEvent
Used by (from the same schema document)	Element failureToPay
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="FailureToPayEvent">
   <xsd:complexContent>
      <xsd:extension base=" CreditEvent "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Language

[Table of contents]

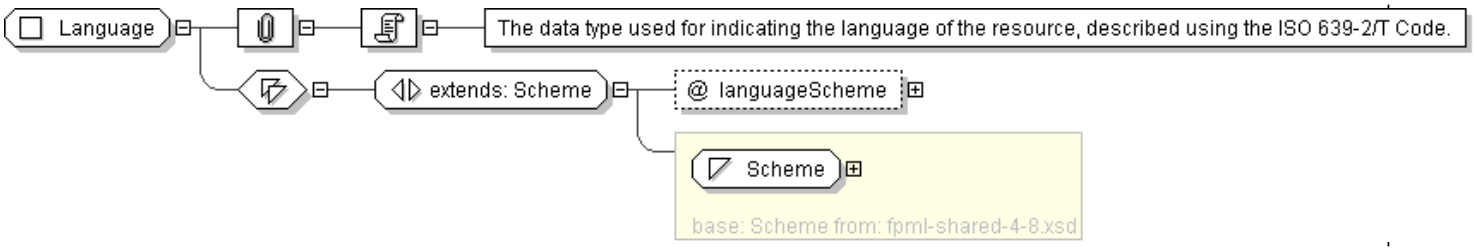
Super-types:	Scheme < Language (by extension)
Sub-types:	None

Name	Language
Used by (from the same schema document)	Complex Type Resource
Abstract	no
Documentation	The data type used for indicating the language of the resource, described using the ISO 639-2/T Code.

XML Instance Representation

```
<...
 languageScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Language">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="languageScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ObligationAccelerationEvent

[Table of contents]

Super-types:	CreditEvent < ObligationAccelerationEvent (by extension)
Sub-types:	None

Name	ObligationAccelerationEvent
Used by (from the same schema document)	Element obligationAcceleration
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="ObligationAccelerationEvent">
   <xsd:complexContent>
      <xsd:extension base=" CreditEvent "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ObligationDefaultEvent

[Table of contents]

Super-types:	CreditEvent < ObligationDefaultEvent (by extension)
Sub-types:	None

Name	ObligationDefaultEvent
Used by (from the same schema document)	Element obligationDefault
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="ObligationDefaultEvent">
   <xsd:complexContent>
      <xsd:extension base=" CreditEvent "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RepudiationMoratoriumEvent

[Table of contents]

Super-types:	CreditEvent < RepudiationMoratoriumEvent (by extension)
Sub-types:	None

Name	RepudiationMoratoriumEvent
Used by (from the same schema document)	Element repudiationMoratorium
Abstract	no

XML Instance Representation

```
<.../>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="RepudiationMoratoriumEvent">
  <xsd:complexContent>
    <xsd:extension base=" CreditEvent "/>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Resource

[Table of contents]

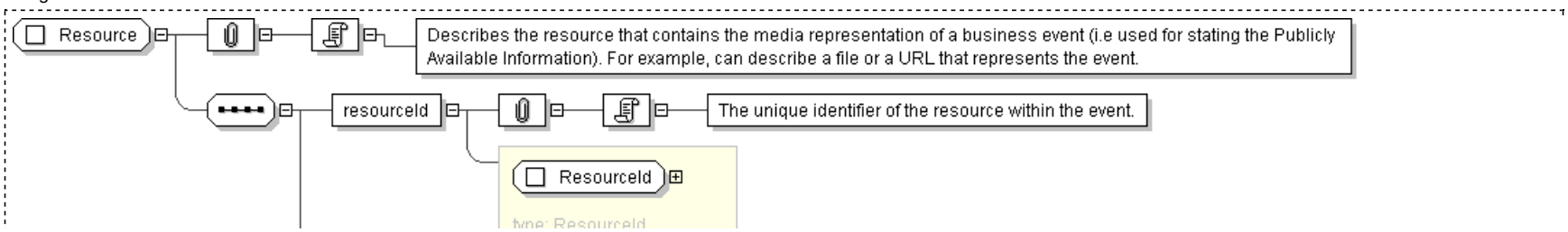
Super-types:	None
Sub-types:	None

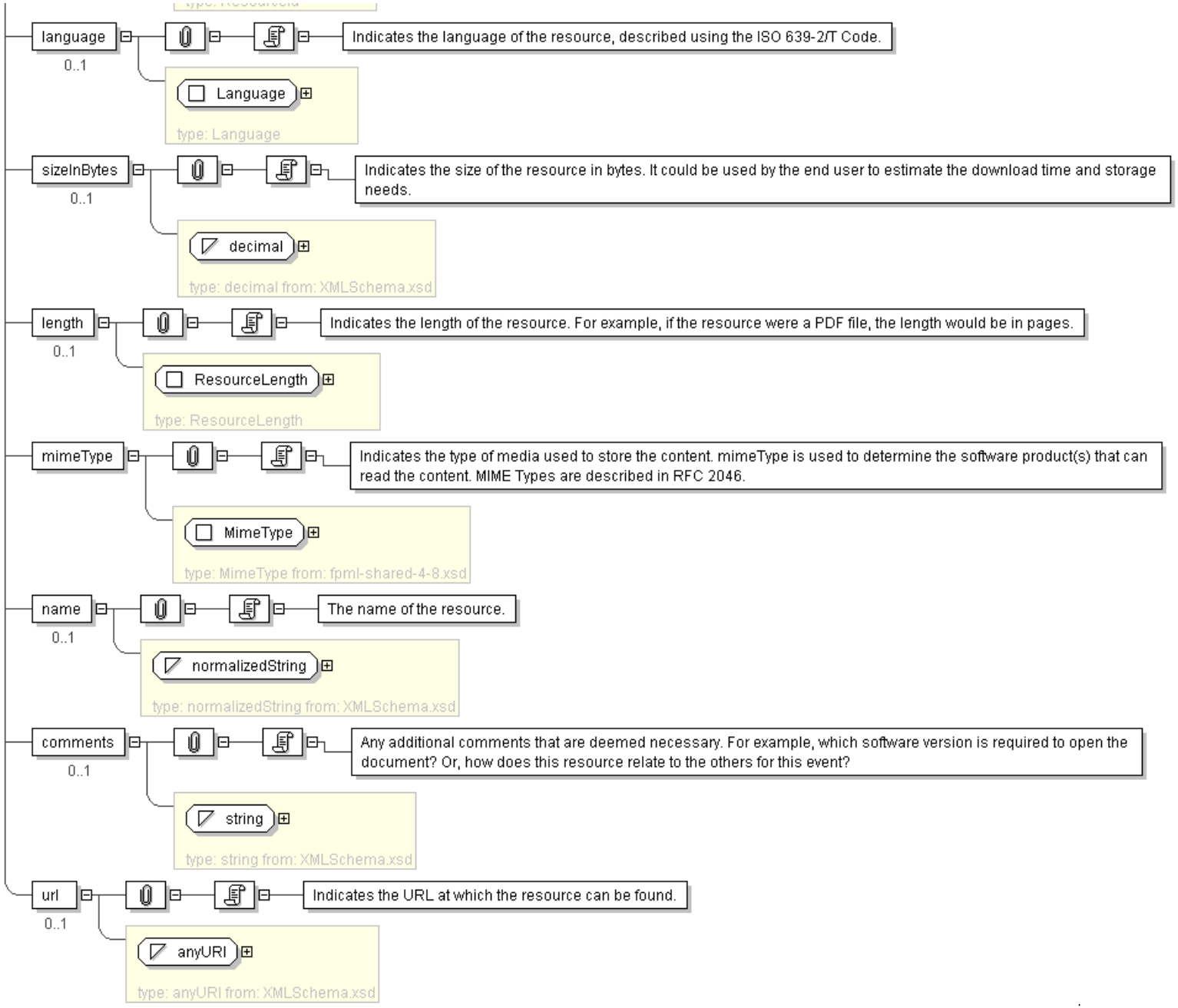
Name	Resource
Used by (from the same schema document)	Complex Type CreditEventNoticeDocument
Abstract	no
Documentation	Describes the resource that contains the media representation of a business event (i.e used for stating the Publicly Available Information). For example, can describe a file or a URL that represents the event.

XML Instance Representation

```
<...>
   <resourceId> ResourceId </resourceId> [1]
    'The unique identifier of the resource within the event.'

   <language> Language </language> [0..1]
    'Indicates the language of the resource, described using the ISO 639-2/T Code.'

   <sizeInBytes> xsd:decimal </sizeInBytes> [0..1]
    'Indicates the size of the resource in bytes. It could be used by the end user to estimate the download time and storage
    needs.'

   <length> ResourceLength </length> [0..1]
    'Indicates the length of the resource. For example, if the resource were a PDF file, the length would be in pages.'

   <mimeType> MimeType </mimeType> [1]
    'Indicates the type of media used to store the content. mimeType is used to determine the software product(s) that can read the
    content. MIME Types are described in RFC 2046.'

   <name> xsd:normalizedString </name> [0..1]
    'The name of the resource.'

   <comments> xsd:string </comments> [0..1]
    'Any additional comments that are deemed necessary. For example, which software version is required to open the document? Or,
    how does this resource relate to the others for this event?'

   <url> xsd:anyURI </url> [0..1]
    'Indicates the URL at which the resource can be found.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="Resource">
  <xsd:sequence>
    <xsd:element name="resourceId" type=" ResourceId "/>
    <xsd:element name="language" type=" Language " minOccurs="0"/>
    <xsd:element name="sizeInBytes" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="length" type=" ResourceLength " minOccurs="0"/>
    <xsd:element name="mimeType" type=" MimeType "/>
```

```
      <xsd:element name="name" type=" xsd:normalizedString " minOccurs="0"/>
      <xsd:element name="comments" type=" xsd:string " minOccurs="0"/>
      <xsd:element name="url" type=" xsd:anyURI " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResourceId

[Table of contents]

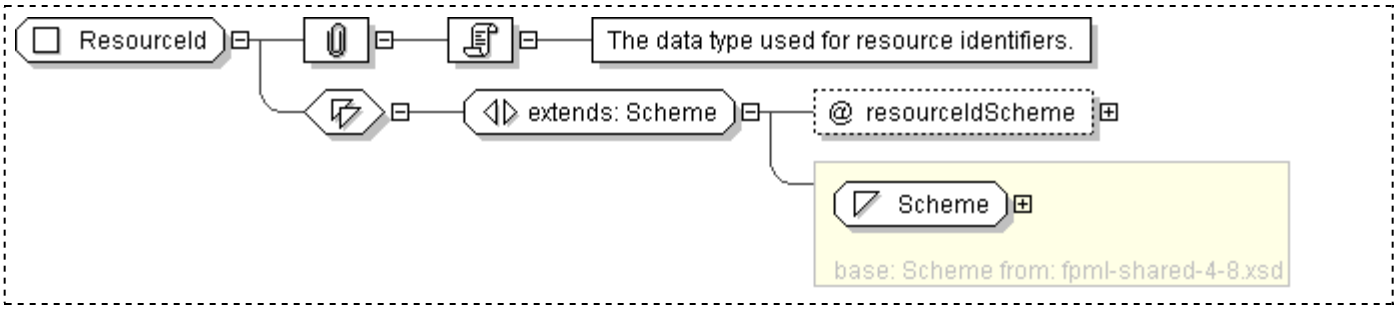
Super-types:	Scheme < ResourceId (by extension)
Sub-types:	None

Name	ResourceId
Used by (from the same schema document)	Complex Type Resource
Abstract	no
Documentation	The data type used for resource identifiers.

XML Instance Representation

```
<...
 resourceIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResourceId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="resourceIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResourceLength

[Table of contents]

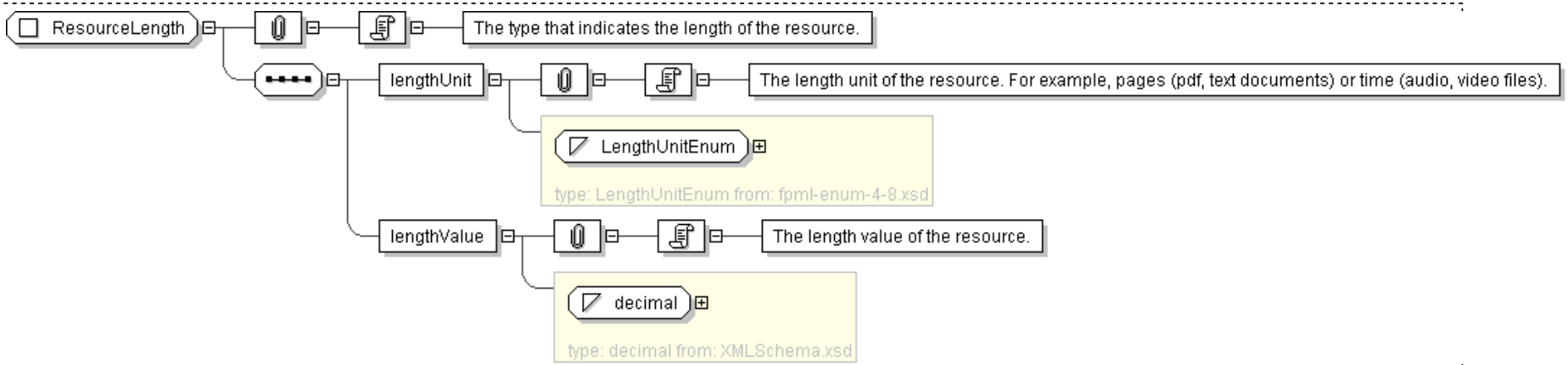
Super-types:	None
Sub-types:	None

Name	ResourceLength
Used by (from the same schema document)	Complex Type Resource
Abstract	no
Documentation	The type that indicates the length of the resource.

XML Instance Representation

```
<...>
   <lengthUnit> LengthUnitEnum </lengthUnit> [1]
   'The length unit of the resource. For example, pages (pdf, text documents) or time (audio, video files).'

   <lengthValue> xsd:decimal </lengthValue> [1]
   'The length value of the resource.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResourceLength">
   <xsd:sequence>
      <xsd:element name="lengthUnit" type=" LengthUnitEnum "/>
      <xsd:element name="lengthValue" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RestructuringEvent

[Table of contents]

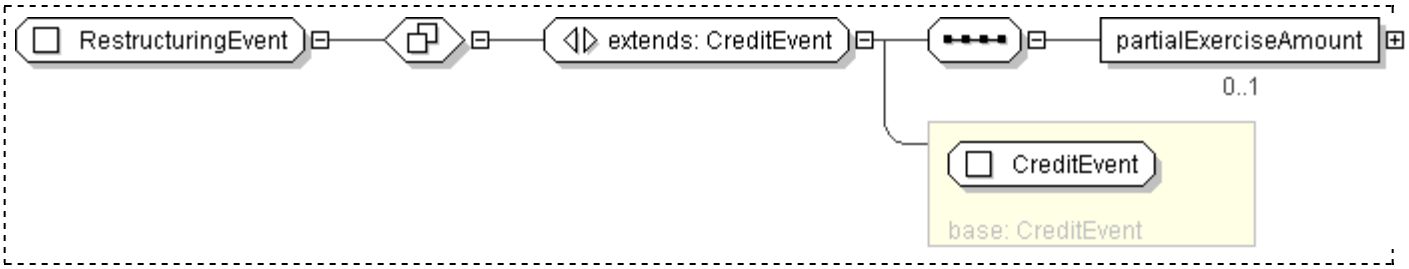
Super-types:	CreditEvent < RestructuringEvent (by extension)
Sub-types:	None

Name	RestructuringEvent
Used by (from the same schema document)	Element restructuring
Abstract	no

XML Instance Representation

```
<...>
   <partialExerciseAmount> Money </partialExerciseAmount> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RestructuringEvent">
   <xsd:complexContent>
      <xsd:extension base=" CreditEvent ">
         <xsd:sequence>
            <xsd:element name="partialExerciseAmount" type=" Money " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: dividendSwapTransactionSupplement
- Global Definitions
  - Complex Type: DividendLeg
  - Complex Type: DividendPeriodPayment
  - Complex Type: DividendSwapTransactionSupplement
  - Complex Type: FixedPaymentAmount
  - Complex Type: FixedPaymentLeg
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 2527 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ○ fpml-eq-shared-4-8.xsd
	  ○ fpml-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 2527 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-eq-shared-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
```

```
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

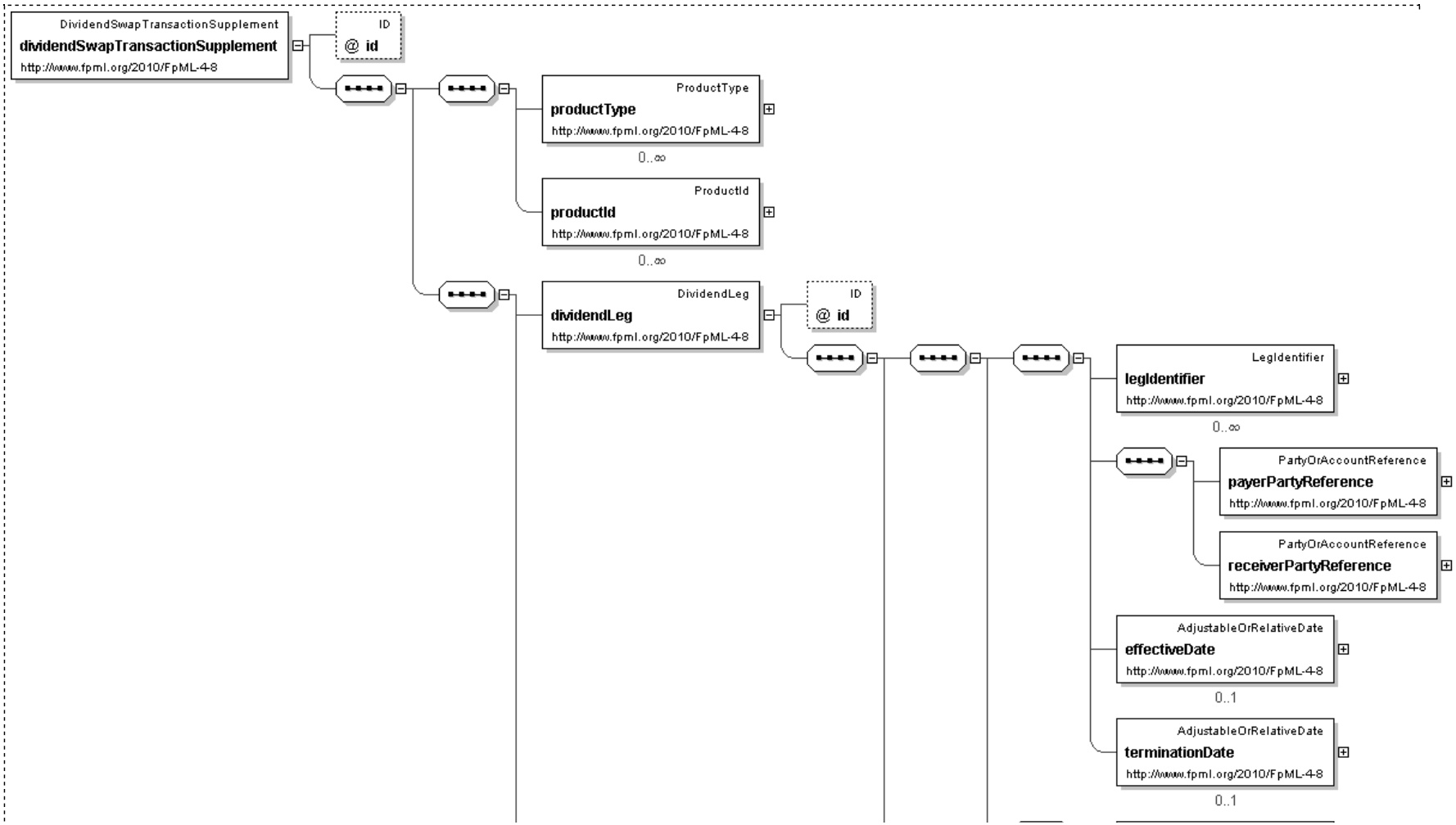
Element: dividendSwapTransactionSupplement

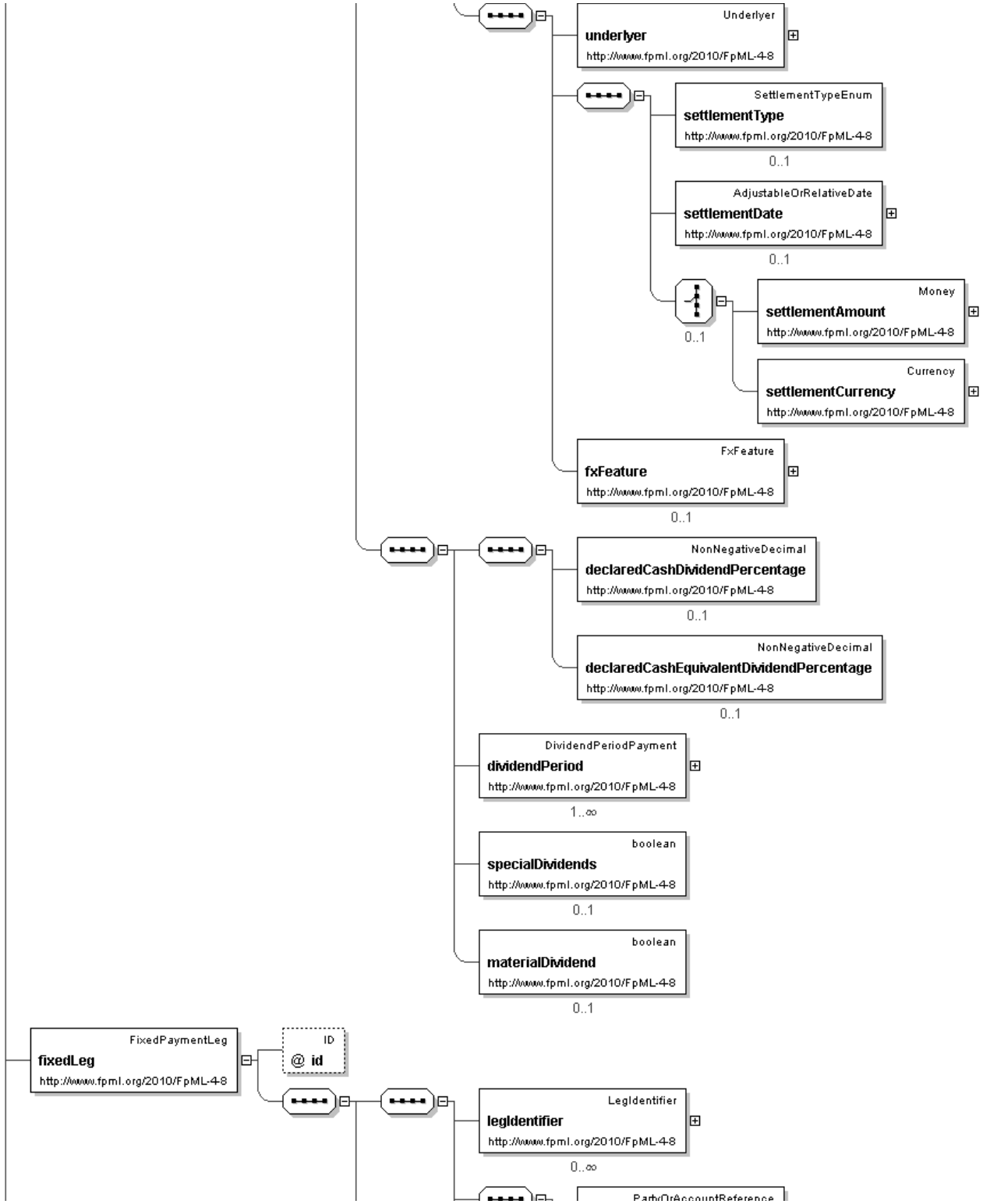
[Table of contents]

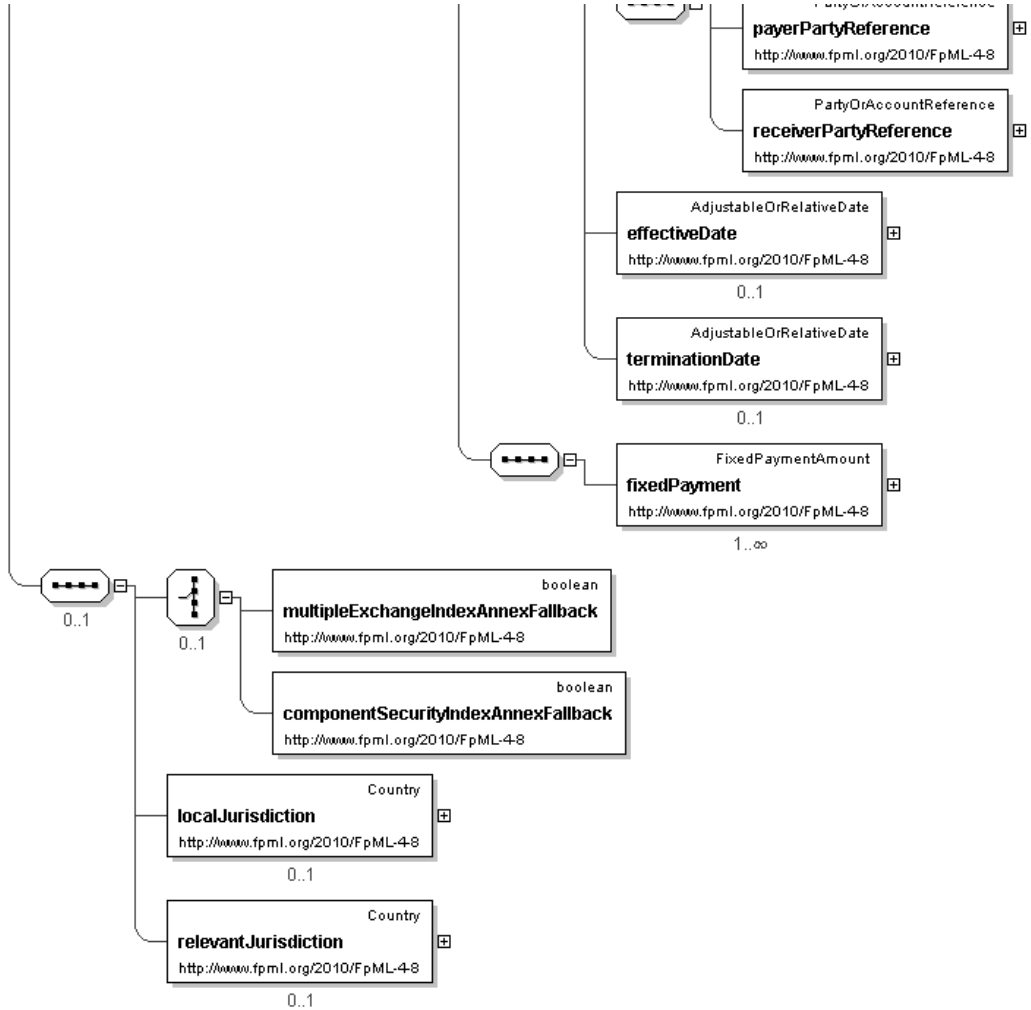
- *This element can be used wherever the following element is referenced:*
  - product

Name	dividendSwapTransactionSupplement
Type	DividendSwapTransactionSupplement
Nillable	no
Abstract	no
Documentation	Specifies the structure of the dividend swap transaction supplement.

Logical Diagram




Underlyer
underlyer
http://www.fpml.org/2010/FpML-4-8
SettlementTypeEnum
settlementType
http://www.fpml.org/2010/FpML-4-8
0..1
AdjustableOrRelativeDate
settlementDate
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
Money
settlementAmount
http://www.fpml.org/2010/FpML-4-8
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
FxFeature
fxFeature
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
declaredCashDividendPercentage
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
declaredCashEquivalentDividendPercentage
http://www.fpml.org/2010/FpML-4-8
0..1
DividendPeriodPayment
dividendPeriod
http://www.fpml.org/2010/FpML-4-8
1..∞
boolean
specialDividends
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
materialDividend
http://www.fpml.org/2010/FpML-4-8
0..1
FixedPaymentLeg
fixedLeg
http://www.fpml.org/2010/FpML-4-8
ID
@ id
LegIdentifier
legIdentifier
http://www.fpml.org/2010/FpML-4-8
0..∞

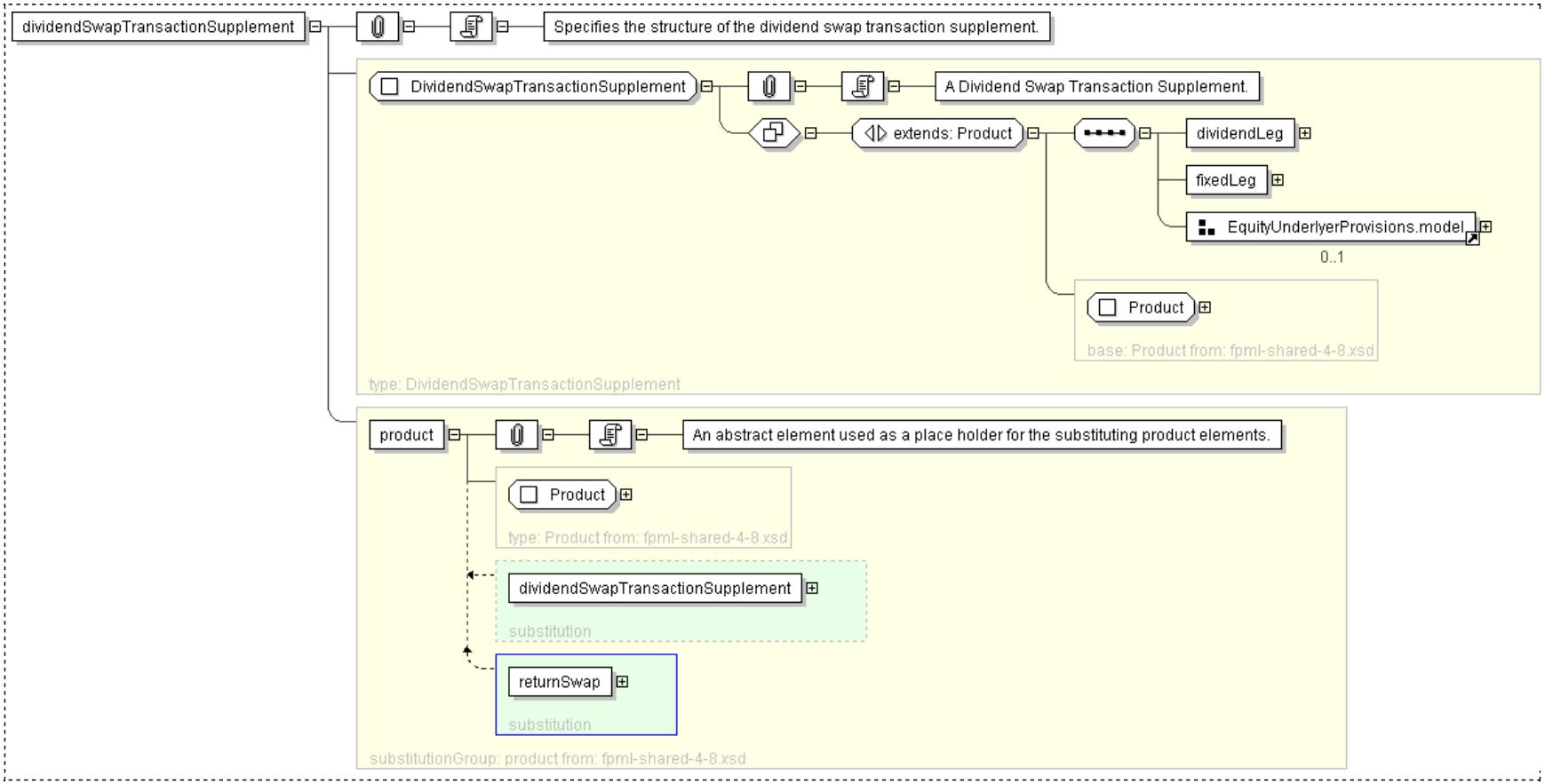


XML Instance Representation

```
<dividendSwapTransactionSupplement
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the
   domain values for this element are not strictly an enumerated list.'

   <dividendLeg> DividendLeg </dividendLeg> [1]
   'Dividend leg.'

   <fixedLeg> FixedPaymentLeg </fixedLeg> [1]
   'Fixed payment leg.'

   Start Group: EquityUnderlyerProvisions.model [0..1]
      Start Group: IndexAnnexFallback.model [0..1]
         Start Choice [1]
            <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
            'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the
            transaction. This annex defines additional provisions which are applicable where an index is comprised of component securities that
            are traded on multiple exchanges.'

            <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
```

```
            'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the
            transaction.'

        End Choice
    End Group: IndexAnnexFallback.model
    <localJurisdiction> Country </localJurisdiction> [0..1]
    'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes, duties,
    and similar charges imposed by the taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction is Not
    Applicable.'

    <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
    'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes,
    duties and similar charges that would be imposed by the taxing authority of the Country of Underlyer on a Hypothetical Broker Dealer
    assuming the Applicable Hedge Positions are held by its office in the Relevant Jurisdiction. If this element is not present Relevant
    Jurisdiction is Not Applicable.'

  End Group: EquityUnderlyerProvisions.model
</dividendSwapTransactionSupplement>
```

Diagram



Schema Component Representation

```
<xsd:element name="dividendSwapTransactionSupplement" type=" DividendSwapTransactionSupplement " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendLeg

[Table of contents]

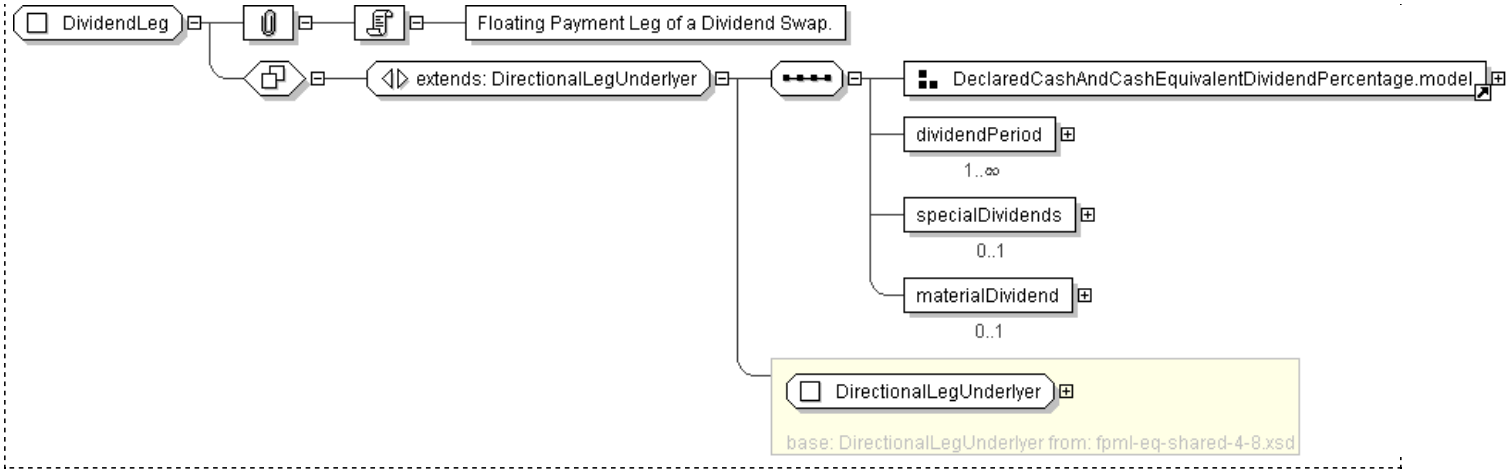
Super-types:	DirectionalLegUnderlyer < DividendLeg (by extension)
Sub-types:	None

Name	DividendLeg
Used by (from the same schema document)	Complex Type DividendSwapTransactionSupplement
Abstract	no
Documentation	Floating Payment Leg of a Dividend Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
    'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
    'Specifies the effective date of this leg of the swap. When defined in relation to a date specified
    somewhere else in the document (through the relativeDate component), this element will typically point to
    the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
    'Specifies the termination date of this leg of the swap. When defined in relation to a date specified
    somewhere else in the document (through the relativeDate component), this element will typically point to
    the termination date of the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlyer of the leg.'

   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
          'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
          'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <fxFeature> FxFeature </fxFeature> [0..1]
    'Quanto, Composite, or Cross Currency FX features.'

   <declaredCashDividendPercentage> NonNegativeDecimal </declaredCashDividendPercentage> [0..1]
    'Declared Cash Dividend Percentage.'

   <declaredCashEquivalentDividendPercentage> NonNegativeDecimal </declaredCashEquivalentDividendPercentage>
   [0..1]
    'Declared Cash Equivalent Dividend Percentage.'

   <dividendPeriod> DividendPeriodPayment </dividendPeriod> [1..*]
    'One to many time bounded dividend payment periods, each with a fixed strike and dividend payment date per
    period.'

   <specialDividends> xsd:boolean </specialDividends> [0..1]
    'If present and true, then special dividends and memorial dividends are applicable.'

   <materialDividend> xsd:boolean </materialDividend> [0..1]
    'If present and true, then material non cash dividends are applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendLeg">
  <xsd:complexContent>
    <xsd:extension base=" DirectionalLegUnderlyer ">
      <xsd:sequence>
        <xsd:group ref=" DeclaredCashAndCashEquivalentDividendPercentage.model "/>
        <xsd:element name="dividendPeriod" type=" DividendPeriodPayment " maxOccurs="unbounded"/>
        <xsd:element name="specialDividends" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="materialDividend" type=" xsd:boolean " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendPeriodPayment

[Table of contents]

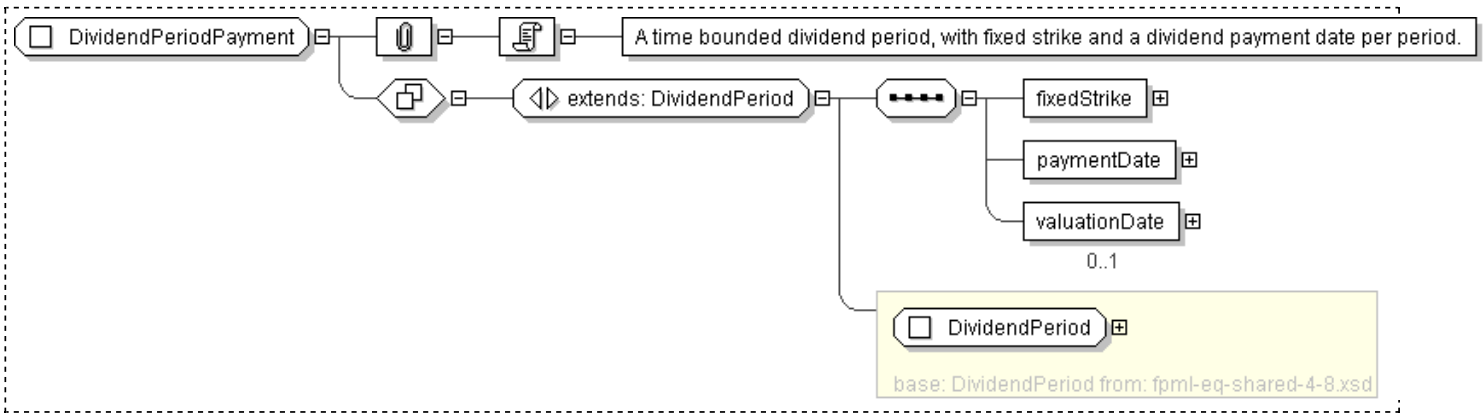
Super-types:	DividendPeriod < DividendPeriodPayment (by extension)
Sub-types:	None

Name	DividendPeriodPayment
Used by (from the same schema document)	Complex Type DividendLeg
Abstract	no
Documentation	A time bounded dividend period, with fixed strike and a dividend payment date per period.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedStartDate> IdentifiedDate </unadjustedStartDate> [1]
   'Unadjusted inclusive dividend period start date.'

   <unadjustedEndDate> IdentifiedDate </unadjustedEndDate> [1]
   'Unadjusted inclusive dividend period end date.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
   'Date adjustments for all unadjusted dates in this dividend period.'

   <underlyerReference> AssetReference </underlyerReference> [0..1]
   'Reference to the underlyer which is paying dividends. This should be used in all cases, and
   must be used where there are multiple underlying assets, to avoid any ambiguity about which
   asset the dividend period relates to.'

   <fixedStrike> PositiveDecimal </fixedStrike> [1]
   'Fixed strike.'

   <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
   'Dividend period amount payment date.'

   <valuationDate> AdjustableOrRelativeDate </valuationDate> [0..1]
   'Dividend period amount valuation date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPeriodPayment">
  <xsd:complexContent>
    <xsd:extension base=" DividendPeriod ">
      <xsd:sequence>
        <xsd:element name="fixedStrike" type=" PositiveDecimal "/>
        <xsd:element name="paymentDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="valuationDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendSwapTransactionSupplement

[Table of contents]

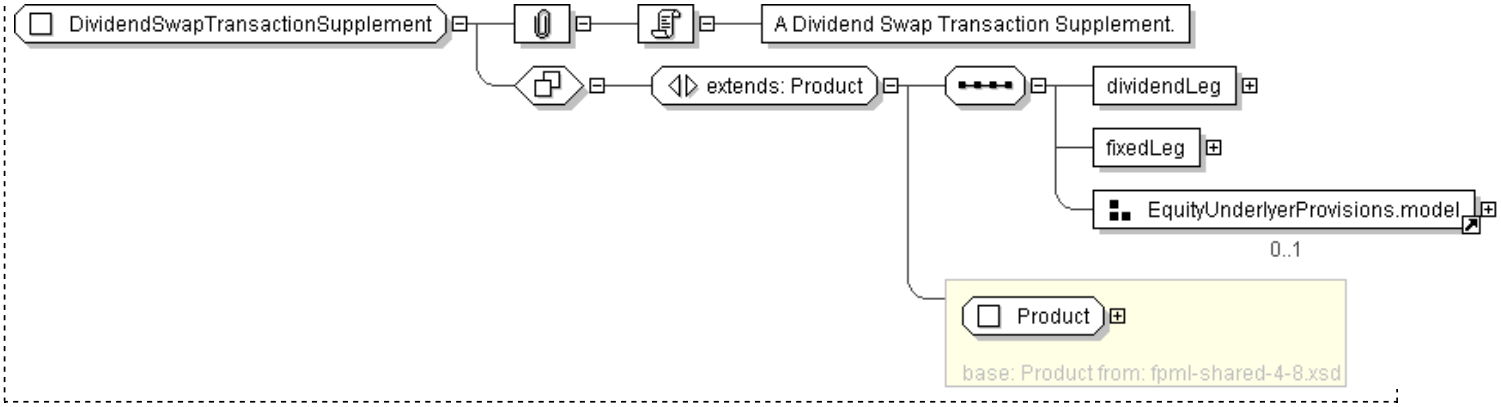
Super-types:	Product < DividendSwapTransactionSupplement (by extension)
Sub-types:	None

Name	DividendSwapTransactionSupplement
Used by (from the same schema document)	Element dividendSwapTransactionSupplement
Abstract	no
Documentation	A Dividend Swap Transaction Supplement.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a
    coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values
    associated with this element. Note that the domain values for this element are not strictly an
    enumerated list.'

   <dividendLeg> DividendLeg </dividendLeg> [1]
    'Dividend leg.'

   <fixedLeg> FixedPaymentLeg </fixedLeg> [1]
    'Fixed payment leg.'

   Start Group: EquityUnderlyerProvisions.model [0..1]
      Start Group: IndexAnnexFallback.model [0..1]
         Start Choice [1]
            <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
             'For an index option transaction, a flag to indicate whether a relevant Multiple
             Exchange Index Annex is applicable to the transaction. This annex defines additional
             provisions which are applicable where an index is comprised of component securities
             that are traded on multiple exchanges.'

            <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback>
            [1]
             'For an index option transaction, a flag to indicate whether a relevant Component
             Security Index Annex is applicable to the transaction.'

         End Choice
      End Group: IndexAnnexFallback.model
      <localJurisdiction> Country </localJurisdiction> [0..1]
       'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to
       determine local taxes, which shall mean taxes, duties, and similar charges imposed by the
       taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction
       is Not Applicable.'

      <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
       'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to
       determine local taxes, which shall mean taxes, duties and similar charges that would be
       imposed by the taxing authority of the Country of Underlyer on a Hypothetical Broker Dealer
       assuming the Applicable Hedge Positions are held by its office in the Relevant
       Jurisdiction. If this element is not present Relevant Jurisdiction is Not Applicable.'

   End Group: EquityUnderlyerProvisions.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendSwapTransactionSupplement">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="dividendLeg" type=" DividendLeg "/>
        <xsd:element name="fixedLeg" type=" FixedPaymentLeg "/>
        <xsd:group ref=" EquityUnderlyerProvisions.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedPaymentAmount

[Table of contents]

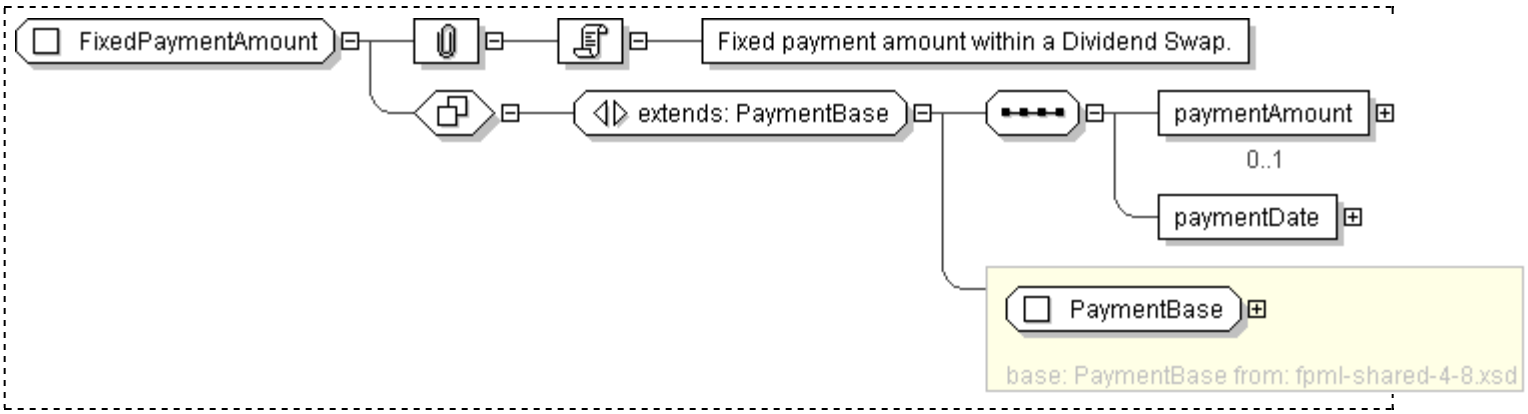
Super-types:	PaymentBase < FixedPaymentAmount (by extension)
Sub-types:	None

Name	FixedPaymentAmount
Used by (from the same schema document)	Complex Type FixedPaymentLeg
Abstract	no
Documentation	Fixed payment amount within a Dividend Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <paymentAmount> Money </paymentAmount> [0..1]
    'Payment amount, which is optional since the payment amount may be calculated
    using fixed strike and number of open units.'

   <paymentDate> RelativeDateOffset </paymentDate> [1]
    'Payment date relative to another date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedPaymentAmount">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
            <xsd:element name="paymentDate" type=" RelativeDateOffset "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FixedPaymentLeg

[Table of contents]

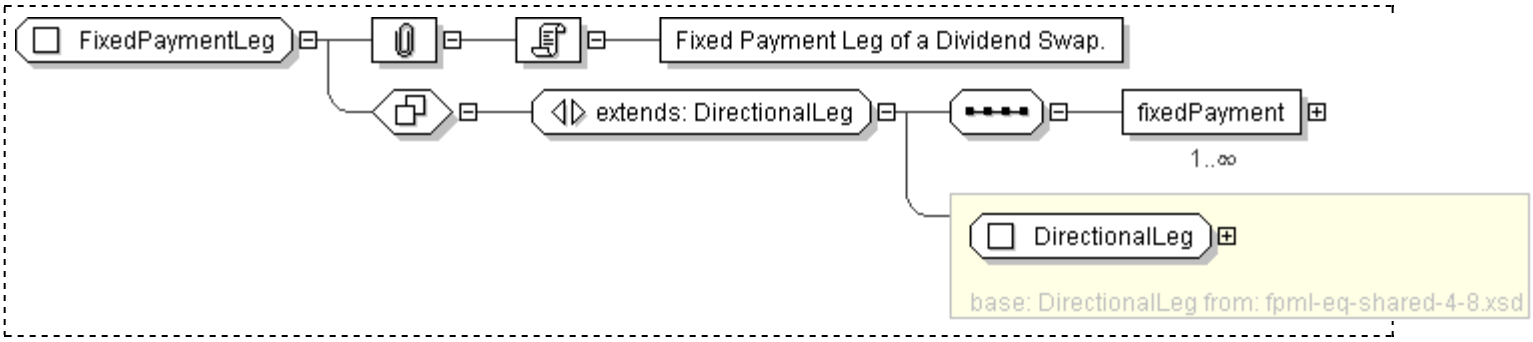
Super-types:	DirectionalLeg < FixedPaymentLeg (by extension)
Sub-types:	None

Name	FixedPaymentLeg
Used by (from the same schema document)	Complex Type DividendSwapTransactionSupplement
Abstract	no
Documentation	Fixed Payment Leg of a Dividend Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
    'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this
    structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this
    structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
    'Specifies the effective date of this leg of the swap. When defined in relation
    to a date specified somewhere else in the document (through the relativeDate
    component), this element will typically point to the effective date of the
    other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
    'Specifies the termination date of this leg of the swap. When defined in
    relation to a date specified somewhere else in the document (through the
    relativeDate component), this element will typically point to the termination
    date of the other leg of the swap.'

   <fixedPayment> FixedPaymentAmount </fixedPayment> [1..*]
    'Fixed payment of a dividend swap, payment date is relative to a dividend
    period payment date. Commonly the dividend leg and the fixed payment leg will
    pay out on the same date, and the payments will be netted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FixedPaymentLeg">
  <xsd:complexContent>
    <xsd:extension base=" DirectionalLeg ">
      <xsd:sequence>
        <xsd:element name="fixedPayment" type=" FixedPaymentAmount "
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: event
  - Element: strategy
- Global Definitions
  - Attribute Group: VersionAttributes.atts
  - Complex Type: Allocation
  - Complex Type: AllocationTradeIdentifier
  - Complex Type: Allocations
  - Complex Type: Amendment
  - Complex Type: Approval
  - Complex Type: Approvals
  - Complex Type: BestFitTrade
  - Complex Type: BlockTradeIdentifier
  - Complex Type: ChangeContract
  - Complex Type: ChangeContractSize
  - Complex Type: Collateral
  - Complex Type: Contract
  - Complex Type: ContractHeader
  - Complex Type: ContractId
  - Complex Type: ContractIdentifier
  - Complex Type: ContractInformation
  - Complex Type: ContractNovation
  - Complex Type: ContractReference
  - Complex Type: ContractTermination
  - Complex Type: CreditDerivativesNotices
  - Complex Type: DataDocument
  - Complex Type: Document
  - Complex Type: Event
  - Complex Type: EventId
  - Complex Type: ExecutionDateTime
  - Complex Type: FirstPeriodStartDate
  - Complex Type: Increase
  - Complex Type: IndependentAmount
  - Complex Type: LinkId
  - Complex Type: PartyPortfolioName
  - Complex Type: PartyRole
  - Complex Type: PartyTradeIdentifier
  - Complex Type: PartyTradeIdentifiers
  - Complex Type: PartyTradeInformation
  - Complex Type: PaymentDetail
  - Complex Type: PaymentRule
  - Complex Type: PercentageRule
  - Complex Type: Portfolio
  - Complex Type: PortfolioName
  - Complex Type: QueryParameter
  - Complex Type: QueryParameterId
  - Complex Type: QueryParameterOperator
  - Complex Type: QueryPortfolio
  - Complex Type: Strategy
  - Complex Type: Trade
  - Complex Type: TradeDifference
  - Complex Type: TradeHeader
  - Complex Type: TradeId

- Complex Type: TradeIdentifier
- Complex Type: TradeSide
- Complex Type: Trader
- Complex Type: Validation
- Complex Type: VersionedContractId
- Complex Type: VersionedTradeId
- Model Group: AccountReferenceOrPartyReference.model
- Model Group: AllocationContent.model
- Model Group: AmendmentDetails.model
- Model Group: CalculationAgent.model
- Model Group: ContractNovationDetails.model
- Model Group: ContractOrContractReference.model
- Model Group: IncreaseDetails.model
- Model Group: TradeOrTradeReference.model
- Model Group: Validation.model
- Simple Type: QueryParameterValue

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 7244 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 7244 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves

as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Attribute Group: VersionAttributes.atts

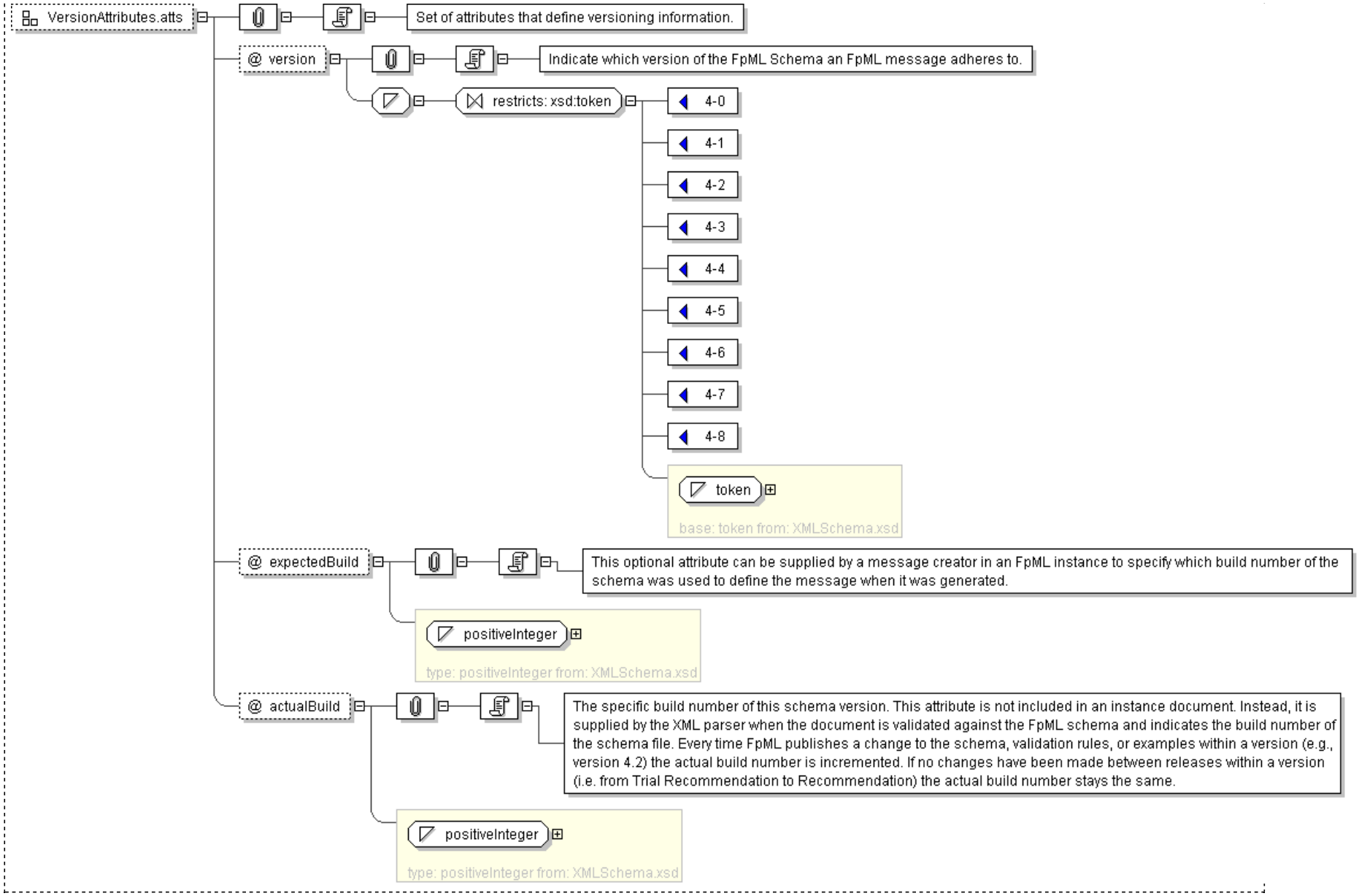
[Table of contents]

Name	VersionAttributes.atts
Used by (from the same schema document)	Complex Type Document
Documentation	Set of attributes that define versioning information.

XML Instance Representation

```
version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

"
expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was
used to define the message when it was generated.'

"
actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied
by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every
time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build
number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to
Recommendation) the actual build number stays the same.'

"
```

Diagram



Schema Component Representation

```
<xsd:attributeGroup name="VersionAttributes.atts">
  <xsd:attribute name="version" use="required">
    <xsd:simpleType>
      <xsd:restriction base=" xsd:token ">
        <xsd:enumeration value="4-0"/>
        <xsd:enumeration value="4-1"/>
        <xsd:enumeration value="4-2"/>
        <xsd:enumeration value="4-3"/>
        <xsd:enumeration value="4-4"/>
        <xsd:enumeration value="4-5"/>
        <xsd:enumeration value="4-6"/>
        <xsd:enumeration value="4-7"/>
```

```
          <xsd:enumeration value="4-8"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="expectedBuild" type=" xsd:positiveInteger "/>
    <xsd:attribute name="actualBuild" type=" xsd:positiveInteger " fixed="3"/>
</xsd:attributeGroup>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: event

[Table of contents]

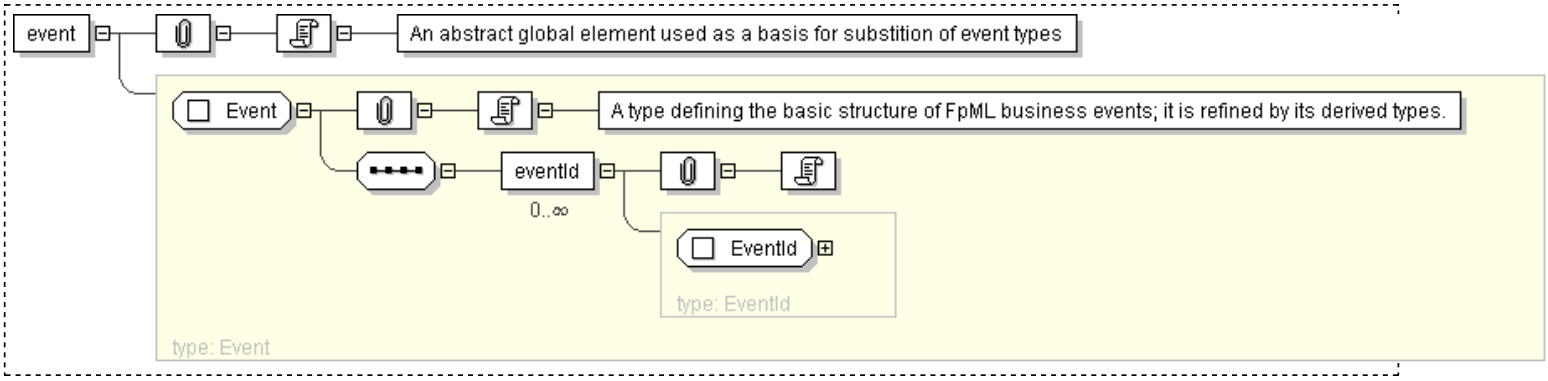
Name	event
Used by (from the same schema document)	Complex Type DataDocument
Type	Event
Nillable	no
Abstract	yes
Documentation	An abstract global element used as a basis for substition of event types

Logical Diagram

XML Instance Representation

```
<event>
   <eventId> EventId </eventId> [0..*]
   ''

</event>
```

Diagram



Schema Component Representation

```
<xsd:element name="event" type=" Event " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

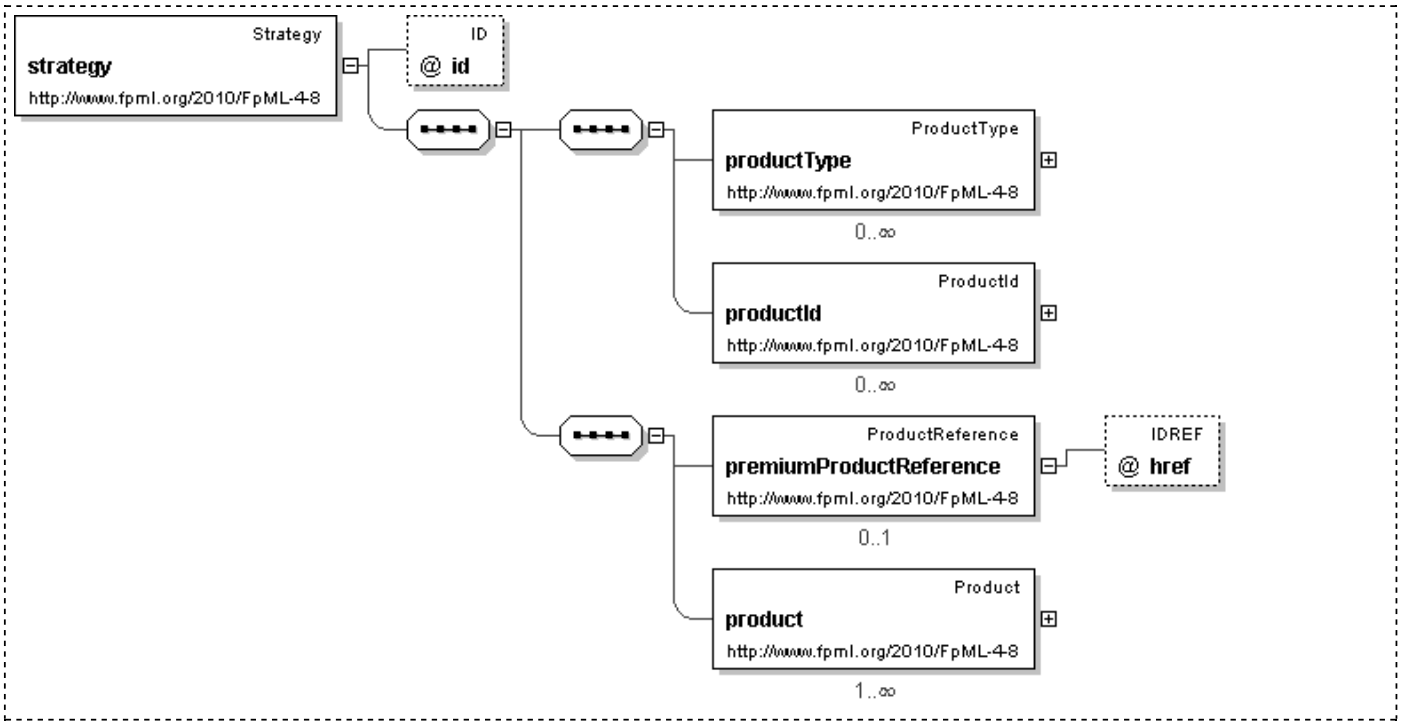
Element: strategy

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	strategy
Type	Strategy
Nillable	no
Abstract	no
Documentation	A strategy product.

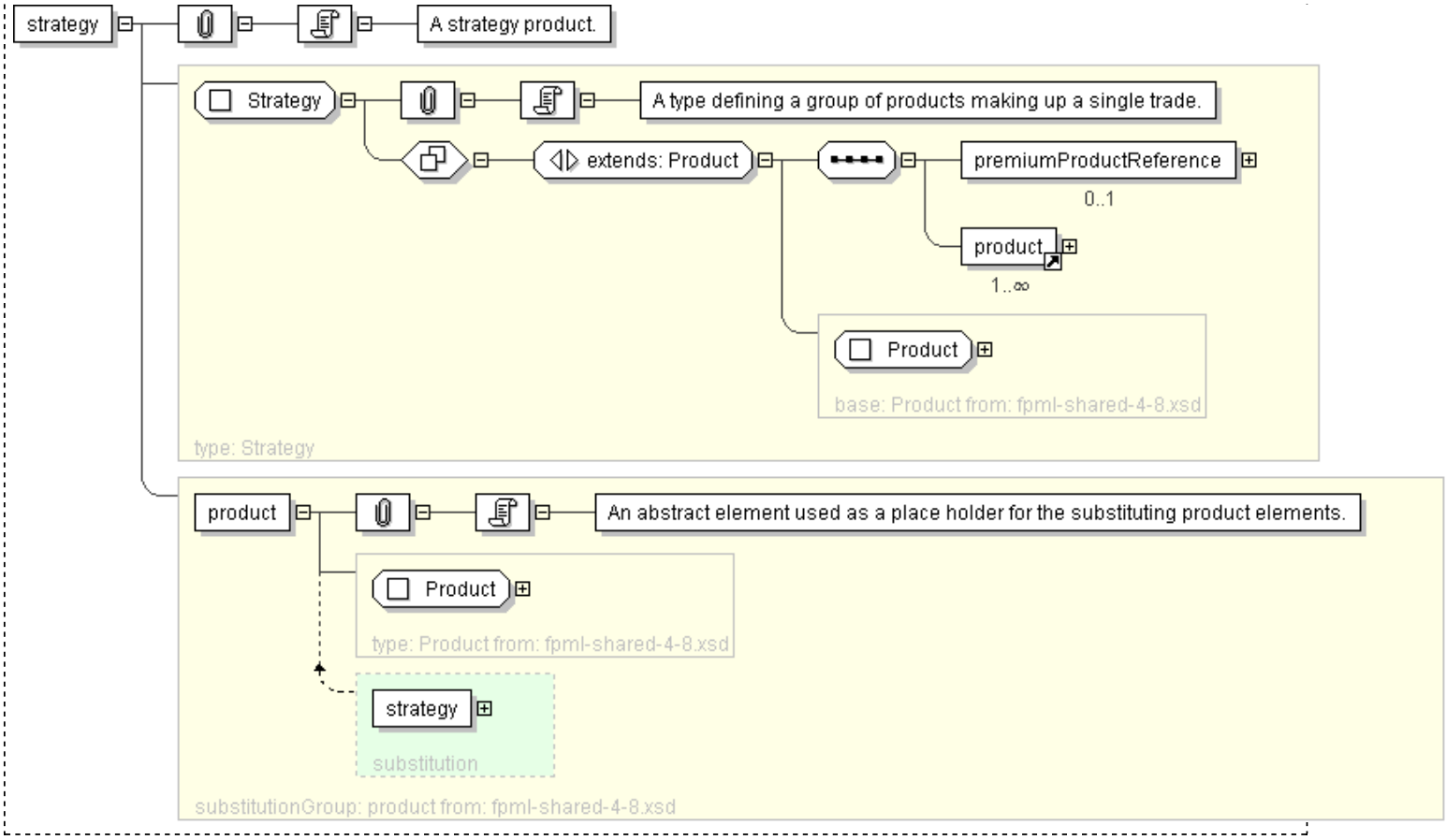
Logical Diagram



XML Instance Representation

```
<strategy
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization
    using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain
    values associated with this element. Note that the domain values for this element are not
    strictly an enumerated list.'

   <premiumProductReference> ProductReference </premiumProductReference> [0..1]
    'Indicates which product within a strategy represents the premium payment.'

   <product> ... </product> [1..*]
</strategy>
```

Diagram



Schema Component Representation

```
<xsd:element name="strategy" type=" Strategy " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AccountReferenceOrPartyReference.model

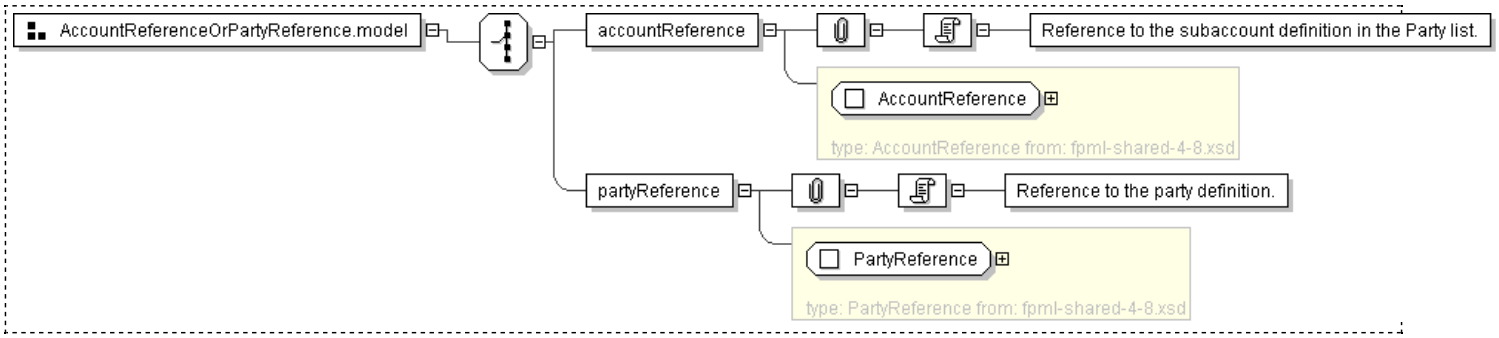
[Table of contents]

Name	AccountReferenceOrPartyReference.model
Used by (from the same schema document)	Complex Type Allocation

XML Instance Representation

```
Start Choice [1]
   <accountReference> AccountReference </accountReference> [1]
   'Reference to the subaccount definition in the Party list.'

   <partyReference> PartyReference </partyReference> [1]
   'Reference to the party definition.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="AccountReferenceOrPartyReference.model">
   <xsd:choice>
      <xsd:element name="accountReference" type=" AccountReference "/>
      <xsd:element name="partyReference" type=" PartyReference "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AllocationContent.model

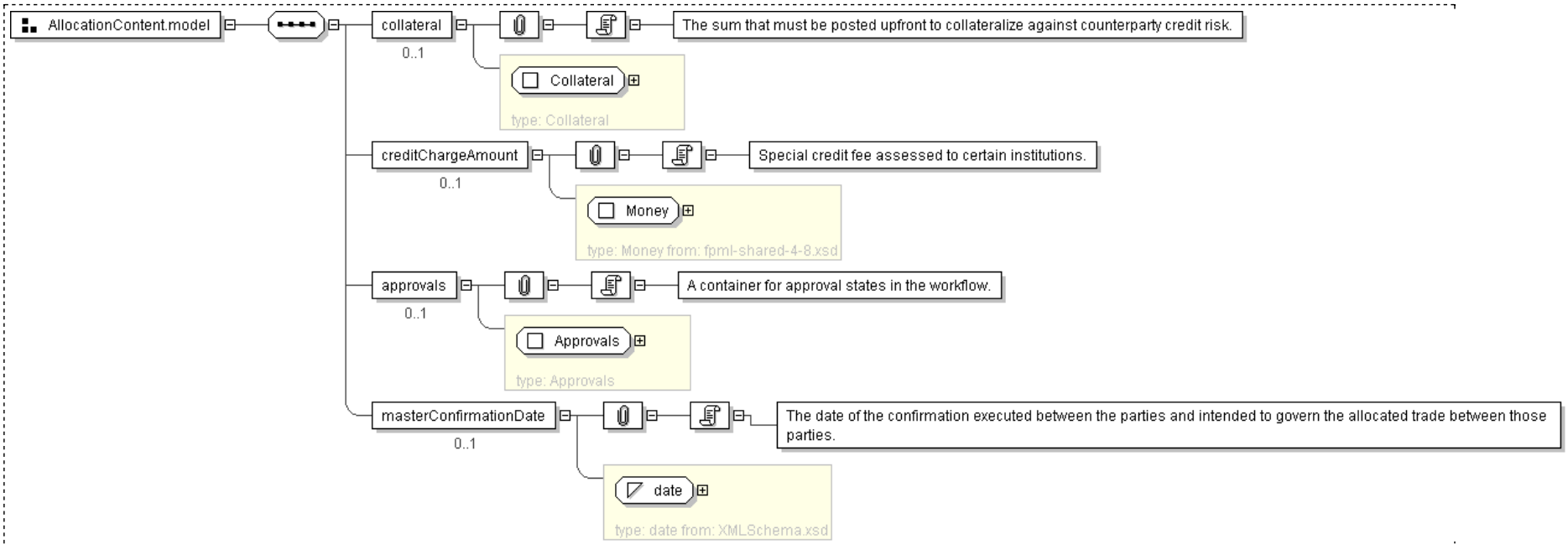
[Table of contents]

Name	AllocationContent.model
Used by (from the same schema document)	Complex Type Allocation

XML Instance Representation

```
<collateral> Collateral </collateral> [0..1]
'The sum that must be posted upfront to collateralize against counterparty credit risk.'

<creditChargeAmount> Money </creditChargeAmount> [0..1]
'Special credit fee assessed to certain institutions.'

<approvals> Approvals </approvals> [0..1]
'A container for approval states in the workflow.'

<masterConfirmationDate> xsd:date </masterConfirmationDate> [0..1]
'The date of the confirmation executed between the parties and intended to govern the allocated trade between those parties.'
```

Diagram



Schema Component Representation

```
<xsd:group name="AllocationContent.model">
  <xsd:sequence>
    <xsd:element name="collateral" type=" Collateral " minOccurs="0"/>
    <xsd:element name="creditChargeAmount" type=" Money " minOccurs="0"/>
    <xsd:element name="approvals" type=" Approvals " minOccurs="0"/>
    <xsd:element name="masterConfirmationDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AmendmentDetails.model

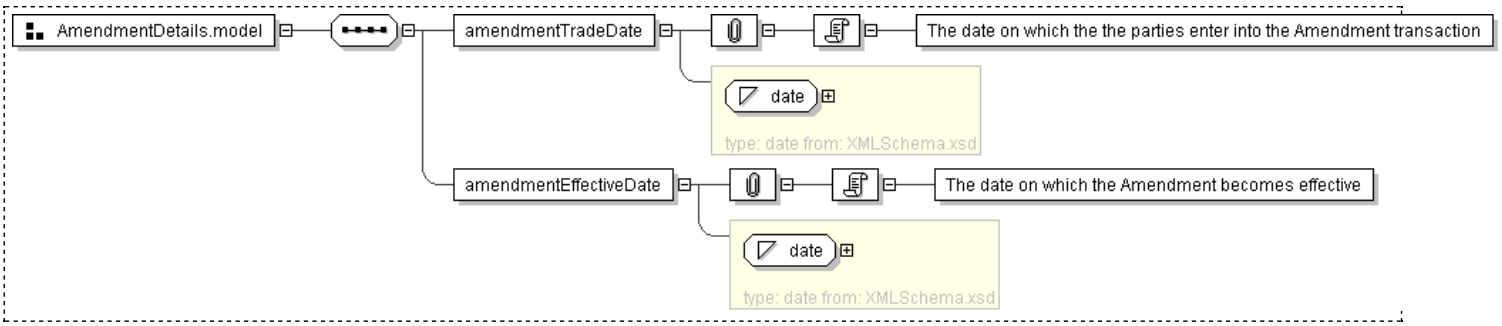
[Table of contents]

Name	AmendmentDetails.model
Used by (from the same schema document)	Complex Type Amendment

XML Instance Representation

```
<amendmentTradeDate> xsd:date </amendmentTradeDate> [1]
'The date on which the the parties enter into the Amendment transaction'

<amendmentEffectiveDate> xsd:date </amendmentEffectiveDate> [1]
'The date on which the Amendment becomes effective'
```

Diagram



Schema Component Representation

```
<xsd:group name="AmendmentDetails.model">
  <xsd:sequence>
    <xsd:element name="amendmentTradeDate" type=" xsd:date "/>
    <xsd:element name="amendmentEffectiveDate" type=" xsd:date "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CalculationAgent.model

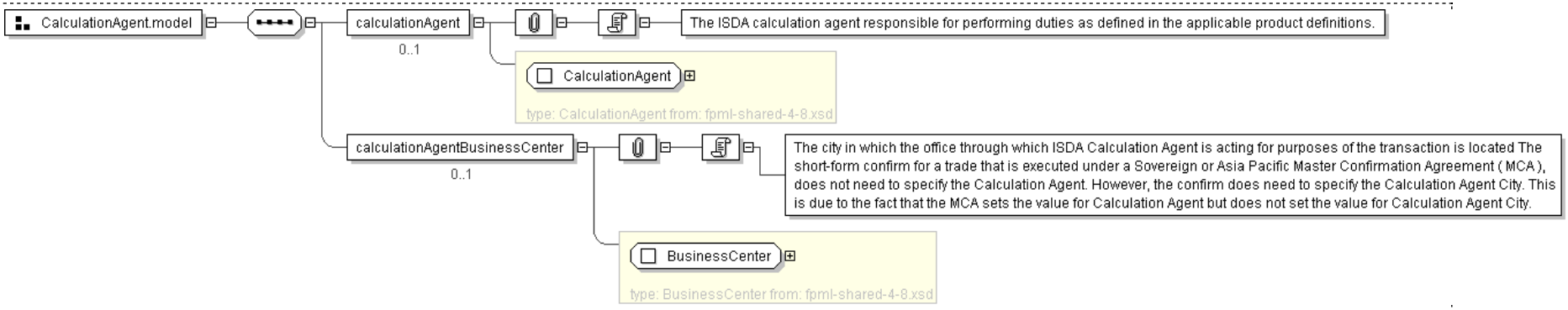
[Table of contents]

Name	CalculationAgent.model
Used by (from the same schema document)	Complex Type Contract , Complex Type Trade

XML Instance Representation

```
<calculationAgent> CalculationAgent </calculationAgent> [0..1]
'The ISDA calculation agent responsible for performing duties as defined in the applicable product definitions.'

<calculationAgentBusinessCenter> BusinessCenter </calculationAgentBusinessCenter> [0..1]
'The city in which the office through which ISDA Calculation Agent is acting for purposes of the transaction is located The short-form confirm for a trade that is executed under a Sovereign or Asia Pacific Master Confirmation Agreement ( MCA ), does not need to specify the Calculation Agent. However, the confirm does need to specify the Calculation Agent City. This is due to the fact that the MCA sets the value for Calculation Agent but does not set the value for Calculation Agent City.'
```

Diagram



Schema Component Representation

```
<xsd:group name="CalculationAgent.model">
  <xsd:sequence>
    <xsd:element name="calculationAgent" type=" CalculationAgent " minOccurs="0"/>
    <xsd:element name="calculationAgentBusinessCenter" type=" BusinessCenter " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: ContractNovationDetails.model

[Table of contents]

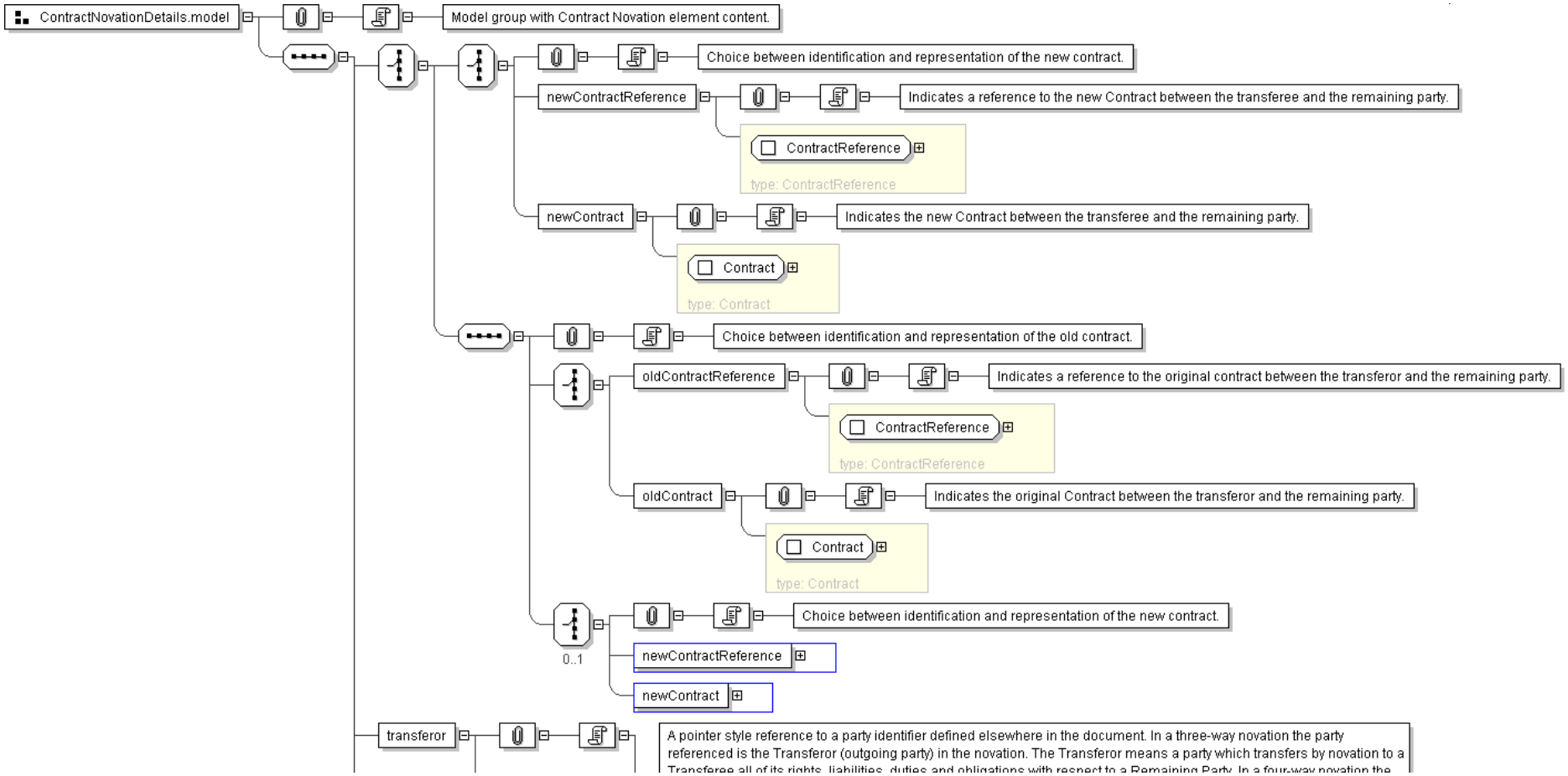
Name	ContractNovationDetails.model
Used by (from the same schema document)	Complex Type ContractNovation
Documentation	Model group with Contract Novation element content.

XML Instance Representation

```
Start Choice [1]
   Start Choice [1]
'Choice between identification and representation of the new contract.'


      <newContractReference> ContractReference </newContractReference> [1]
      'Indicates a reference to the new Contract between the transferee and the remaining party.'

      <newContract> Contract </newContract> [1]
      'Indicates the new Contract between the transferee and the remaining party.'

   End Choice
   Start Choice [1]
      <oldContractReference> ContractReference </oldContractReference> [1]
      'Indicates a reference to the original contract between the transferor and the remaining party.'

      <oldContract> Contract </oldContract> [1]
      'Indicates the original Contract between the transferor and the remaining party.'

   End Choice
   Start Choice [0..1]
'Choice between identification and representation of the new contract.'


      <newContractReference> ContractReference </newContractReference> [1]
      <newContract> Contract </newContract> [1]
   End Choice
End Choice
<transferor> PartyReference </transferor> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferor (outgoing
party) in the novation. The Transferor means a party which transfers by novation to a Transferee all of its rights, liabilities, duties and obligations with
respect to a Remaining Party. In a four-way novation the party referenced is Transferor 1 which transfers by novation to Transferee 1 all of its rights,
liabilities, duties and obligations with respect to Transferor 2. ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-way
novation).'

<transferee> PartyReference </transferee> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferee (incoming
party) in the novation. Transferee means a party which accepts by way of novation all rights, liabilities, duties and obligations of a Transferor with respect
to a Remaining Party. In a four-way novation the party referenced is Transferee 1 which accepts by way of novation the rights, liabilities, duties and
obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).'

<remainingParty> PartyReference </remainingParty> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the
novation. Remaining Party means a party which consents to a Transferor\'s transfer by novation and the acceptance thereof by the Transferee of all of the
Transferor\'s rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In
a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to
Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation)
or Transferor 2 (four-way novation).'

<otherRemainingParty> PartyReference </otherRemainingParty> [0..1]
'A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be
omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities,
duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).'

<novationDate> xsd:date </novationDate> [1]
'Specifies the date that one party\'s legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of
the 2004 ISDA Novation Definitions, section 1.16.'

<novationContractDate> xsd:date </novationContractDate> [0..1]
'Specifies the date the parties agree to assign or novate a Contract. If this element is not specified, the novationContractDate will be deemed to be the
novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.'

Start Choice [1]
'Choice for expressing the novated amount as either a money amount, number of options, or number of units, according the the financial product which is being
novated.'


   <novatedAmount> Money </novatedAmount> [1]
   'The amount which represents the portion of the Old Contract being novated.'
```

```
<novatedNumberOfOptions> xsd:decimal </novatedNumberOfOptions> [1]
'The number of options which represent the portion of the Old Contract being novated.'

<novatedNumberOfUnits> xsd:decimal </novatedNumberOfUnits> [1]
'The number of options which represent the portion of the Old Contract being novated.'

End Choice
<fullFirstCalculationPeriod> xsd:boolean </fullFirstCalculationPeriod> [0..1]
'This element corresponds to the applicability of the Full First Calculation Period as defined in the 2004 ISDA Novation Definitions, section 1.20.'

<firstPeriodStartDate> FirstPeriodStartDate </firstPeriodStartDate> [0..2]
'Element that is used to be able to make sense of the â€œnew transactionâ€ without requiring reference back to the â€œold transactionâ€. In the case of interest rate products there are potentially 2 â€œfirst period start datesâ€ to reference â€“ one with respect to each party to the new transaction. For Credit Default Swaps there is just the one with respect to the party that is the fixed rate payer.'

<nonReliance> Empty </nonReliance> [0..1]
'This element corresponds to the non-Reliance section in the 2004 ISDA Novation Definitions, section 2.1 (c) (i). The element appears in the instance document when non-Reliance is applicable.'

<creditDerivativesNotices> CreditDerivativesNotices </creditDerivativesNotices> [0..1]
'This element should be specified if one or more of either a Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party. The type of notice or notices that have been delivered should be indicated by setting the relevant boolean element value(s) to true. The absence of the element means that no Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party.'

<contractualDefinitions> ContractualDefinitions </contractualDefinitions> [0..*]
'The definitions (such as those published by ISDA) that will define the terms of the novation transaction.'

<contractualTermsSupplement> ContractualTermsSupplement </contractualTermsSupplement> [0..*]
'A contractual supplement (such as those published by ISDA) that will apply to the trade.'
```

Diagram



party referenced is Transferor 1 which transfers by novation to Transferee 1 all of its rights, liabilities, duties and obligations with respect to Transferor 2. ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

transferee

A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferee (incoming party) in the novation. Transferee means a party which accepts by way of novation all rights, liabilities, duties and obligations of a Transferor with respect to a Remaining Party. In a four-way novation the party referenced is Transferee 1 which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

remainingParty

A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the novation. Remaining Party means a party which consents to a Transferor's transfer by novation and the acceptance thereof by the Transferee of all of the Transferor's rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation) or Transferor 2 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

otherRemainingParty
0..1

A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

novationDate

Specifies the date that one party's legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of the 2004 ISDA Novation Definitions, section 1.16.

date
type: date from: XMLSchema.xsd

novationContractDate
0..1

Specifies the date the parties agree to assign or novate a Contract. If this element is not specified, the novationContractDate will be deemed to be the novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.

date
type: date from: XMLSchema.xsd

Choice for expressing the novated amount as either a money amount, number of options, or number of units, according the the financial product which is being novated.

novatedAmount

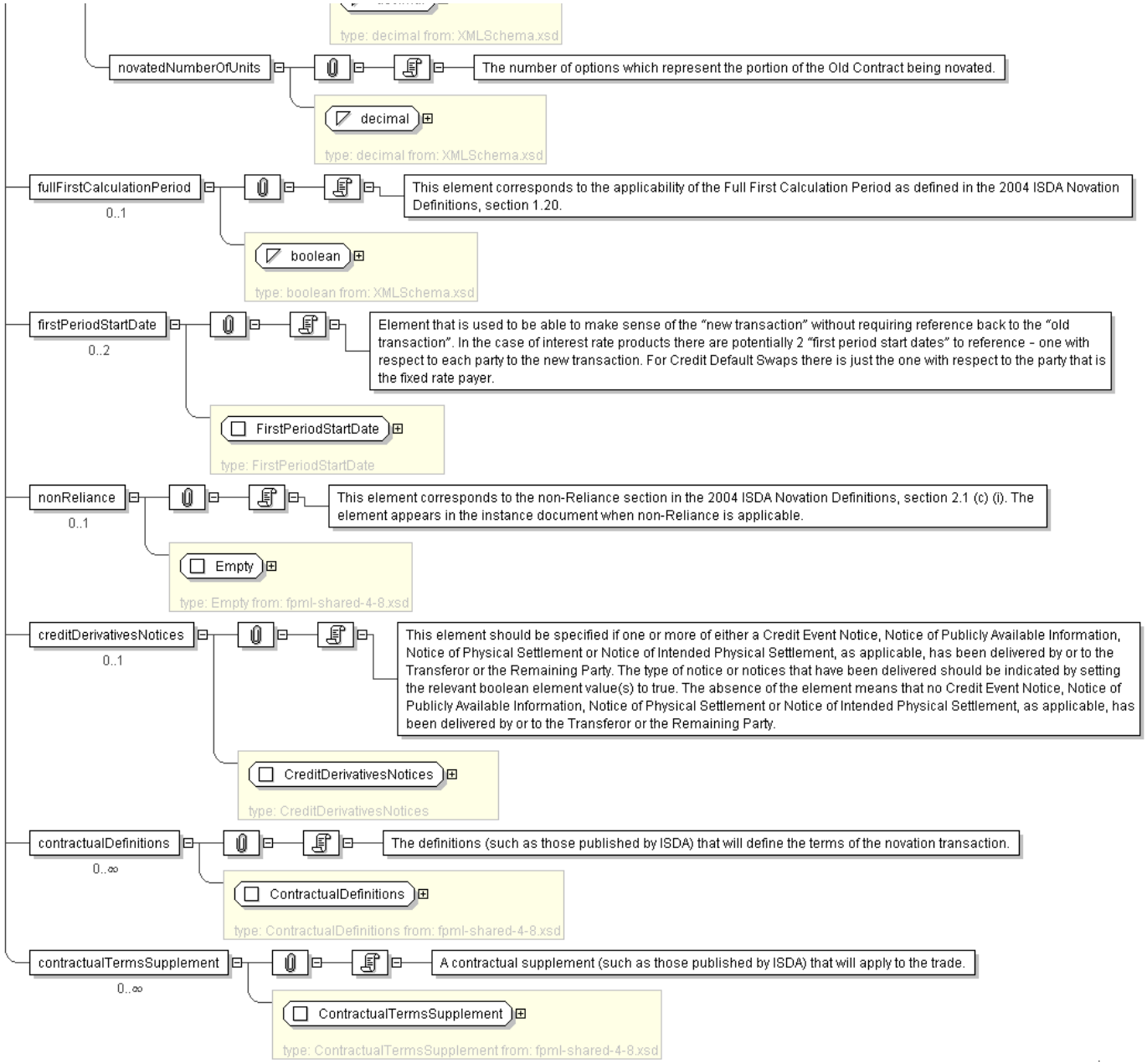
The amount which represents the portion of the Old Contract being novated.

Money
type: Money from: fpml-shared-4-8.xsd

novatedNumberOfOptions

The number of options which represent the portion of the Old Contract being novated.

decimal



Schema Component Representation

```
<xsd:group name="ContractNovationDetails.model">
  <xsd:sequence>
    <xsd:choice>
      <xsd:choice>
        <xsd:element name="newContractReference" type=" ContractReference "/>
        <xsd:element name="newContract" type=" Contract "/>
      </xsd:choice>
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="oldContractReference" type=" ContractReference "/>
          <xsd:element name="oldContract" type=" Contract "/>
        </xsd:choice>
```

```
          <xsd:choice minOccurs="0">
             <xsd:element name="newContractReference" type=" ContractReference "/>
             <xsd:element name="newContract" type=" Contract "/>
          </xsd:choice>
       </xsd:sequence>
    </xsd:choice>
    <xsd:element name="transferor" type=" PartyReference "/>
    <xsd:element name="transferee" type=" PartyReference "/>
    <xsd:element name="remainingParty" type=" PartyReference "/>
    <xsd:element name="otherRemainingParty" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="novationDate" type=" xsd:date "/>
    <xsd:element name="novationContractDate" type=" xsd:date " minOccurs="0"/>
    <xsd:choice>
       <xsd:element name="novatedAmount" type=" Money "/>
       <xsd:element name="novatedNumberOfOptions" type=" xsd:decimal "/>
       <xsd:element name="novatedNumberOfUnits" type=" xsd:decimal "/>
    </xsd:choice>
    <xsd:element name="fullFirstCalculationPeriod" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="firstPeriodStartDate" type=" FirstPeriodStartDate " minOccurs="0" maxOccurs="2"/>
    <xsd:element name="nonReliance" type=" Empty " minOccurs="0"/>
    <xsd:element name="creditDerivativesNotices" type=" CreditDerivativesNotices " minOccurs="0"/>
    <xsd:element name="contractualDefinitions" type=" ContractualDefinitions " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="contractualTermsSupplement" type=" ContractualTermsSupplement " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: ContractOrContractReference.model

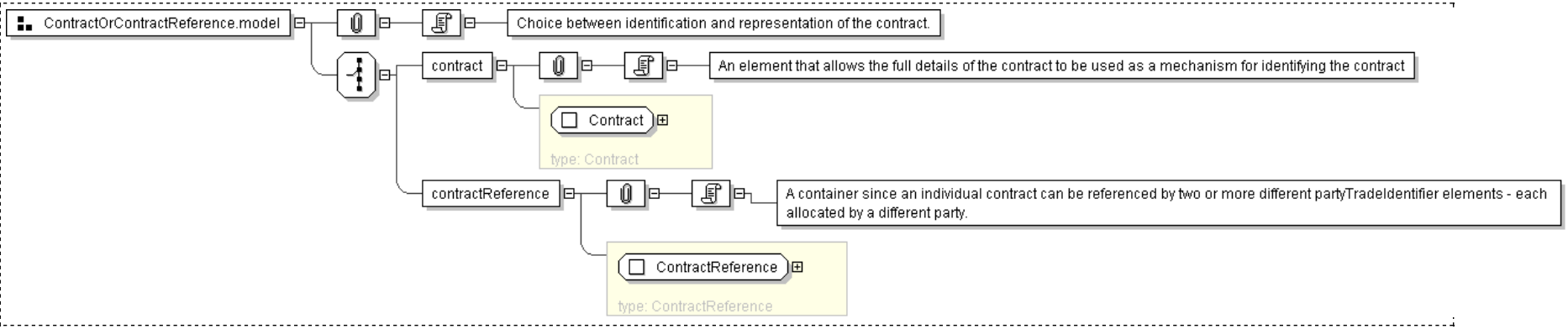
[Table of contents]

Name	ContractOrContractReference.model
Documentation	Choice between identification and representation of the contract.

XML Instance Representation

```
Start Choice [1]
   <contract> Contract </contract> [1]
   'An element that allows the full details of the contract to be used as a mechanism for identifying the contract'

   <contractReference> ContractReference </contractReference> [1]
   'A container since an individual contract can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different
   party.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="ContractOrContractReference.model">
  <xsd:choice>
    <xsd:element name="contract" type=" Contract "/>
    <xsd:element name="contractReference" type=" ContractReference "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: IncreaseDetails.model

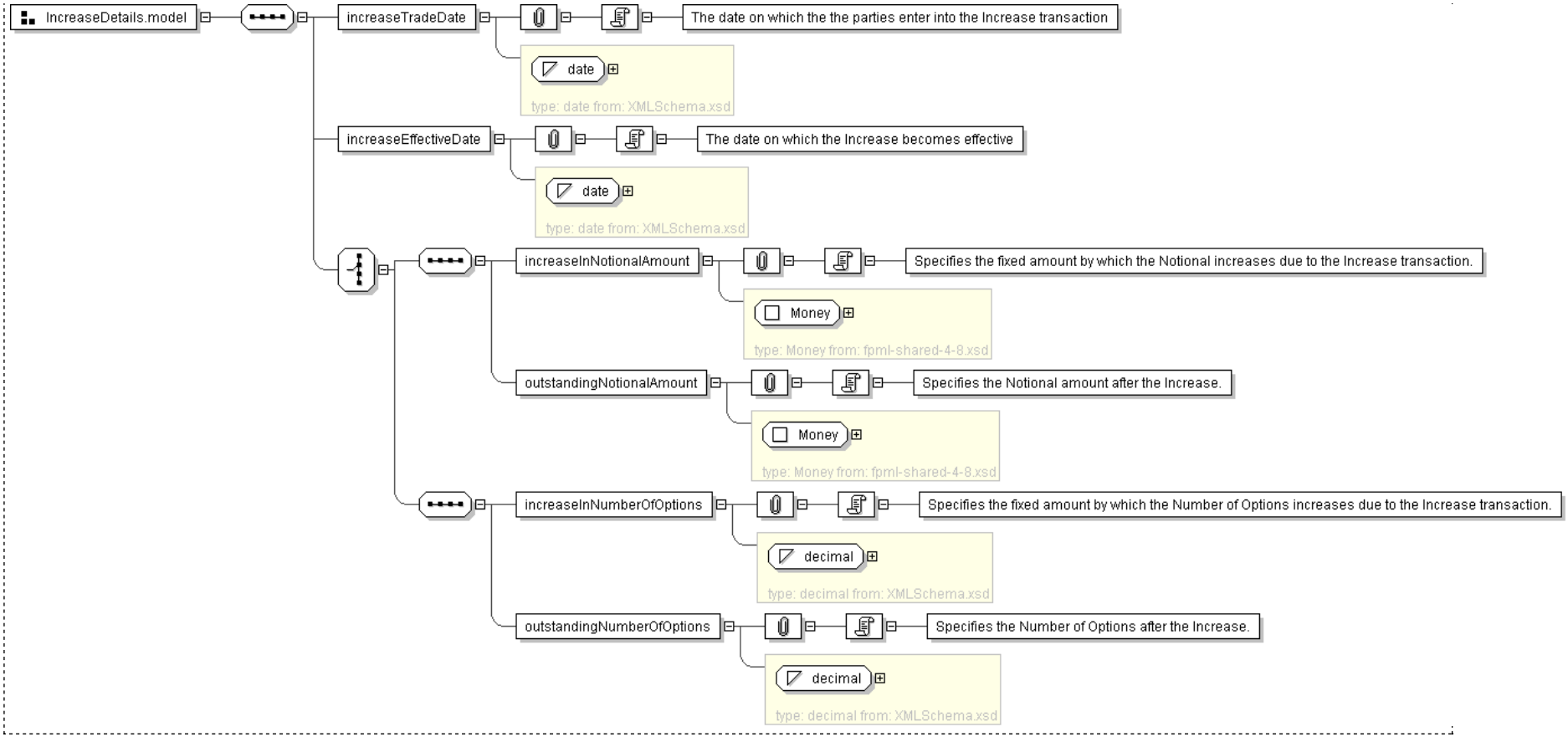
[Table of contents]

Name	IncreaseDetails.model
Used by (from the same schema document)	Complex Type Increase

XML Instance Representation

```
<increaseTradeDate> xsd:date </increaseTradeDate> [1]
'The date on which the the parties enter into the Increase transaction'

<increaseEffectiveDate> xsd:date </increaseEffectiveDate> [1]
'The date on which the Increase becomes effective'

Start Choice [1]
   <increaseInNotionalAmount> Money </increaseInNotionalAmount> [1]
   'Specifies the fixed amount by which the Notional increases due to the Increase transaction.'

   <outstandingNotionalAmount> Money </outstandingNotionalAmount> [1]
   'Specifies the Notional amount after the Increase.'

   <increaseInNumberOfOptions> xsd:decimal </increaseInNumberOfOptions> [1]
   'Specifies the fixed amount by which the Number of Options increases due to the Increase transaction.'

   <outstandingNumberOfOptions> xsd:decimal </outstandingNumberOfOptions> [1]
   'Specifies the Number of Options after the Increase.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="IncreaseDetails.model">
  <xsd:sequence>
    <xsd:element name="increaseTradeDate" type=" xsd:date "/>
    <xsd:element name="increaseEffectiveDate" type=" xsd:date "/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="increaseInNotionalAmount" type=" Money "/>
        <xsd:element name="outstandingNotionalAmount" type=" Money "/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="increaseInNumberOfOptions" type=" xsd:decimal "/>
        <xsd:element name="outstandingNumberOfOptions" type=" xsd:decimal "/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: TradeOrTradeReference.model

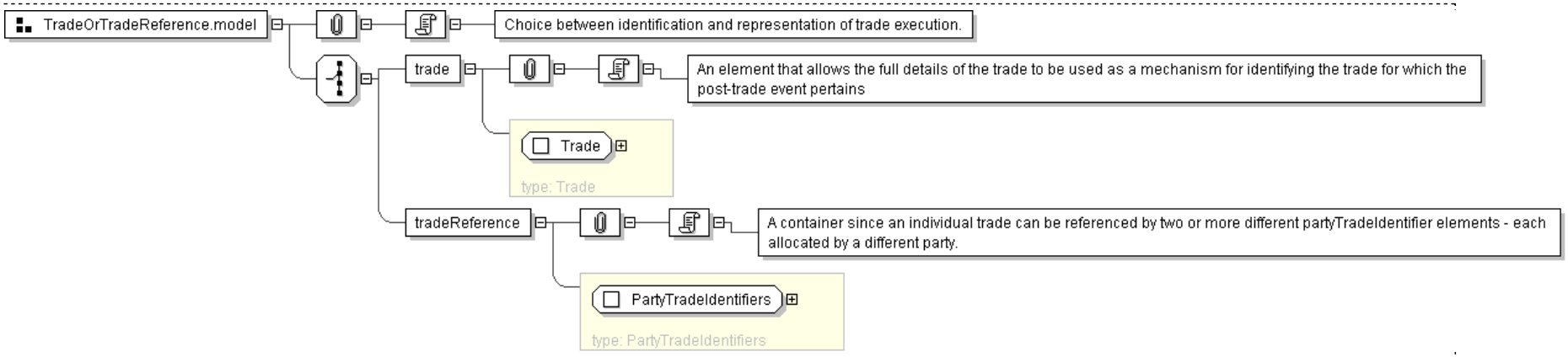
[Table of contents]

Name	TradeOrTradeReference.model
Used by (from the same schema document)	Complex Type Increase
Documentation	Choice between identification and representation of trade execution.

XML Instance Representation

```
Start Choice [1]
   <trade> Trade </trade> [1]
   'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event
   pertains'

   <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
   'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a
   different party.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="TradeOrTradeReference.model">
   <xsd:choice>
      <xsd:element name="trade" type=" Trade "/>
      <xsd:element name="tradeReference" type=" PartyTradeIdentifiers "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Validation.model

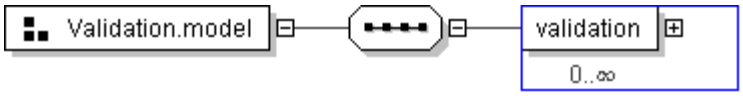
[Table of contents]

Name	Validation.model
Used by (from the same schema document)	Complex Type DataDocument

XML Instance Representation

```
<validation> Validation </validation> [0..*]
```

Diagram



Schema Component Representation

```
<xsd:group name="Validation.model">
  <xsd:sequence>
    <xsd:element name="validation" type=" Validation " minOccurs="0"
    maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Allocation

[Table of contents]

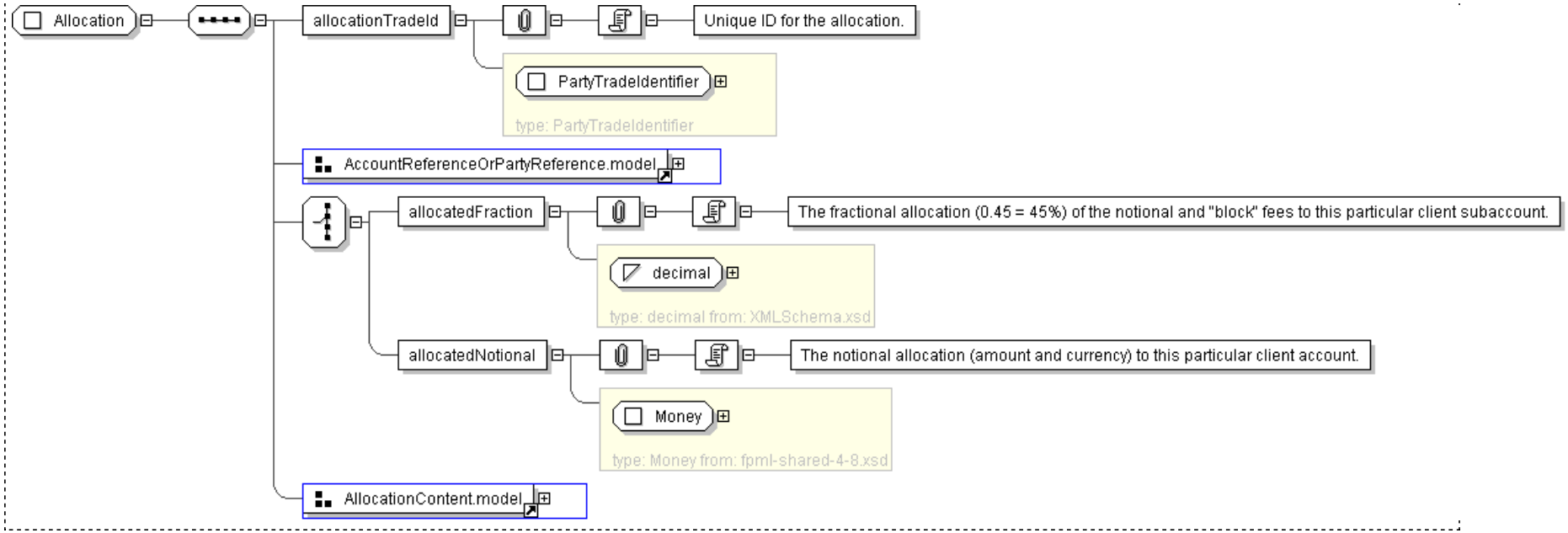
Super-types:	None
Sub-types:	None

Name	Allocation
Used by (from the same schema document)	Complex Type Allocations
Abstract	no

XML Instance Representation

```
<...>
   <allocationTradeId> PartyTradeIdentifier </allocationTradeId> [1]
   'Unique ID for the allocation.'

   Start Choice [1]
      <accountReference> AccountReference </accountReference> [1]
      'Reference to the subaccount definition in the Party list.'

      <partyReference> PartyReference </partyReference> [1]
      'Reference to the party definition.'

   End Choice
   Start Choice [1]
      <allocatedFraction> xsd:decimal </allocatedFraction> [1]
      'The fractional allocation (0.45 = 45%) of the notional and \"block\" fees to this particular client subaccount.'

      <allocatedNotional> Money </allocatedNotional> [1]
      'The notional allocation (amount and currency) to this particular client account.'

   End Choice
   <collateral> Collateral </collateral> [0..1]
   'The sum that must be posted upfront to collateralize against counterparty credit risk.'

   <creditChargeAmount> Money </creditChargeAmount> [0..1]
   'Special credit fee assessed to certain institutions.'

   <approvals> Approvals </approvals> [0..1]
   'A container for approval states in the workflow.'

   <masterConfirmationDate> xsd:date </masterConfirmationDate> [0..1]
   'The date of the confirmation executed between the parties and intended to govern the allocated trade between those parties.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Allocation">
  <xsd:sequence>
    <xsd:element name="allocationTradeId" type=" PartyTradeIdentifier "/>
    <xsd:group ref=" AccountReferenceOrPartyReference.model "/>
    <xsd:choice>
      <xsd:element name="allocatedFraction" type=" xsd:decimal "/>
      <xsd:element name="allocatedNotional" type=" Money "/>
    </xsd:choice>
    <xsd:group ref=" AllocationContent.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Allocations

[Table of contents]

Super-types:	None
Sub-types:	None

Name	Allocations
Used by (from the same schema document)	Complex Type Trade
Abstract	no

XML Instance Representation

```
<...>
   <allocation> Allocation </allocation> [1..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="Allocations">
   <xsd:sequence>
      <xsd:element name="allocation" type=" Allocation " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AllocationTradeIdentifier

[Table of contents]

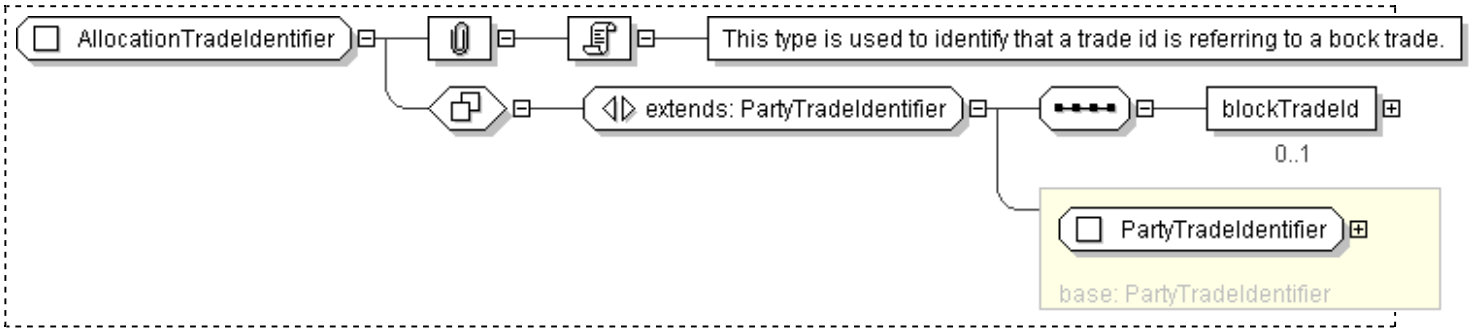
Super-types:	TradeIdentifier < PartyTradeIdentifier (by extension) < AllocationTradeIdentifier (by extension)
Sub-types:	None

Name	AllocationTradeIdentifier
Abstract	no
Documentation	This type is used to identify that a trade id is referring to a bock trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the
   document. The party referenced has allocated the trade identifier.'

   Start Choice [1..*]
      <tradeId> TradeId </tradeId> [1]
      <versionedTradeId> VersionedTradeId </versionedTradeId> [1]
   End Choice
   <linkId> LinkId </linkId> [0..*]
   'A link identifier allowing the trade to be associated with other related trades,
   e.g. the linkId may contain a tradeId for an associated trade or several related
   trades may be given the same linkId. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not
   strictly an enumerated list.'

   <blockTradeId> PartyTradeIdentifier </blockTradeId> [0..1]
   'The trade id of the block trade. This is used by each one of the allocated trades
   to reference the block trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AllocationTradeIdentifier">
   <xsd:complexContent>
      <xsd:extension base=" PartyTradeIdentifier ">
         <xsd:sequence>
            <xsd:element name="blockTradeId" type=" PartyTradeIdentifier " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Amendment

[Table of contents]

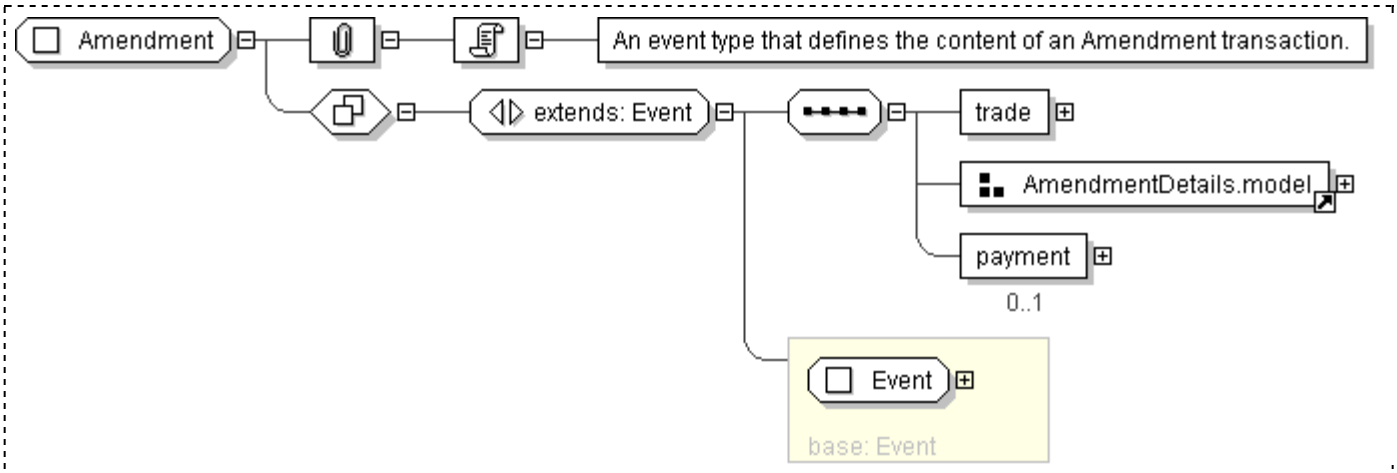
Super-types:	Event < Amendment (by extension)
Sub-types:	None

Name	Amendment
Abstract	no
Documentation	An event type that defines the content of an Amendment transaction.

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

   <trade> Trade </trade> [1]
   <amendmentTradeDate> xsd:date </amendmentTradeDate> [1]
   'The date on which the the parties enter into the Amendment transaction'

   <amendmentEffectiveDate> xsd:date </amendmentEffectiveDate> [1]
   'The date on which the Amendment becomes effective'

   <payment> Payment </payment> [0..1]
   'A payment for the right to amend the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Amendment">
   <xsd:complexContent>
      <xsd:extension base=" Event ">
         <xsd:sequence>
            <xsd:element name="trade" type=" Trade "/>
            <xsd:group ref=" AmendmentDetails.model "/>
            <xsd:element name="payment" type=" Payment " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Approval

[Table of contents]

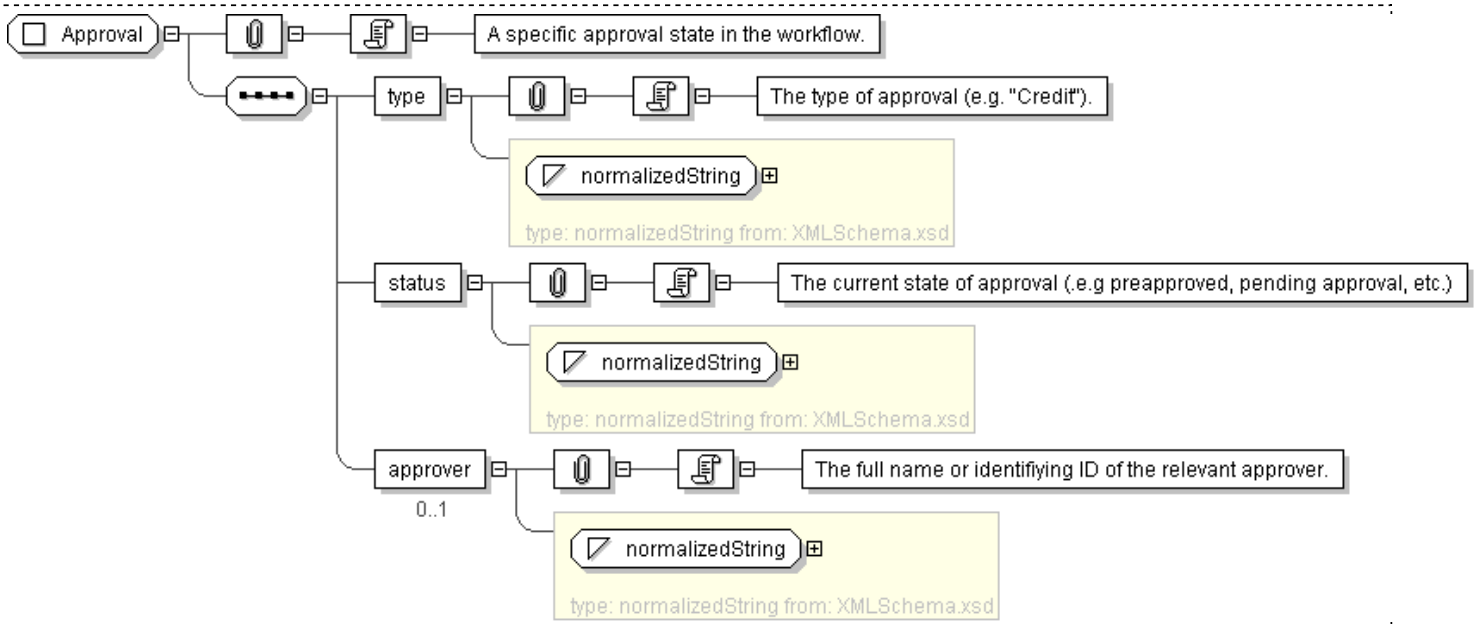
Super-types:	None
Sub-types:	None

Name	Approval
Used by (from the same schema document)	Complex Type Approvals
Abstract	no
Documentation	A specific approval state in the workflow.

XML Instance Representation

```
<...>
   <type> xsd:normalizedString </type> [1]
   'The type of approval (e.g. \"Credit\").'

   <status> xsd:normalizedString </status> [1]
   'The current state of approval (.e.g preapproved, pending approval, etc.)'

   <approver> xsd:normalizedString </approver> [0..1]
   'The full name or identifiying ID of the relevant approver.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Approval">
   <xsd:sequence>
      <xsd:element name="type" type=" xsd:normalizedString "/>
      <xsd:element name="status" type=" xsd:normalizedString "/>
      <xsd:element name="approver" type=" xsd:normalizedString " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Approvals

[Table of contents]

Super-types:	None
Sub-types:	None

Name	Approvals
Used by (from the same schema document)	Model Group AllocationContent.model
Abstract	no

XML Instance Representation

```
<...>
   <approval> Approval </approval> [1..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="Approvals">
   <xsd:sequence>
      <xsd:element name="approval" type=" Approval " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BestFitTrade

[Table of contents]

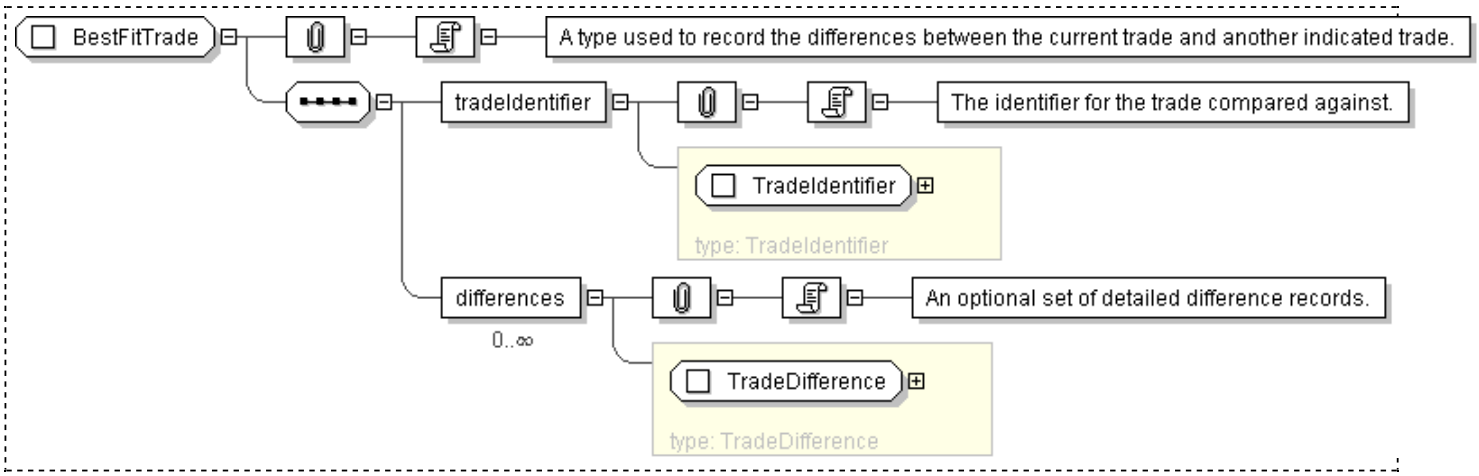
Super-types:	None
Sub-types:	None

Name	BestFitTrade
Abstract	no
Documentation	A type used to record the differences between the current trade and another indicated trade.

XML Instance Representation

```
<...>
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'The identifier for the trade compared against.'

   <differences> TradeDifference </differences> [0..*]
   'An optional set of detailed difference records.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BestFitTrade">
   <xsd:sequence>
      <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
      <xsd:element name="differences" type=" TradeDifference " minOccurs="0"
      maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BlockTradeIdentifier

[Table of contents]

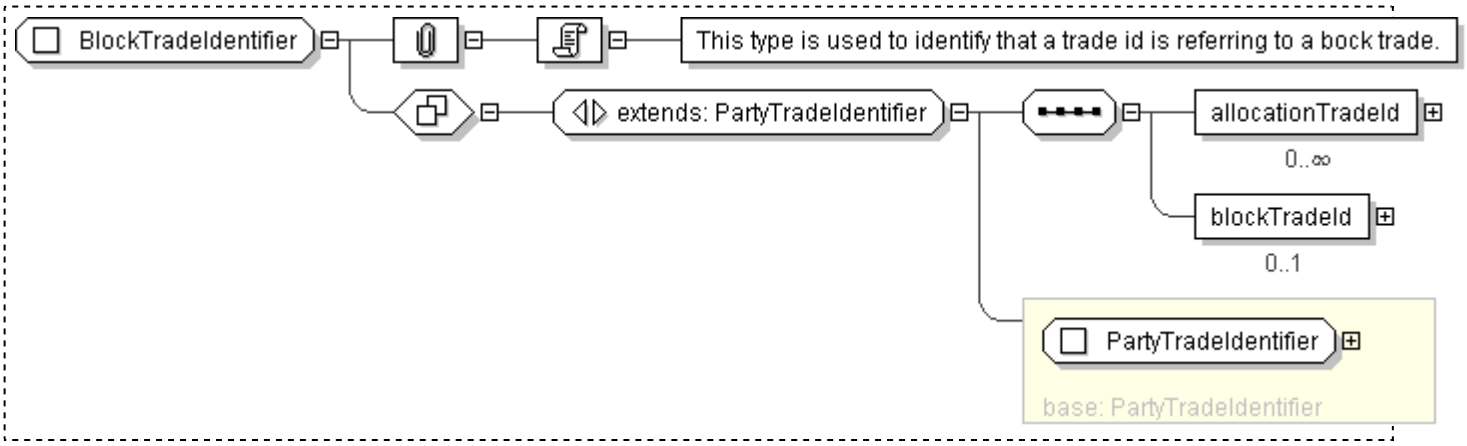
Super-types:	TradeIdentifier < PartyTradeIdentifier (by extension) < BlockTradeIdentifier (by extension)
Sub-types:	None

Name	BlockTradeIdentifier
Abstract	no
Documentation	This type is used to identify that a trade id is referring to a bock trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the
   document. The party referenced has allocated the trade identifier.'

   Start Choice [1..*]
      <tradeId> TradeId </tradeId> [1]
      <versionedTradeId> VersionedTradeId </versionedTradeId> [1]
   End Choice
   <linkId> LinkId </linkId> [0..*]
   'A link identifier allowing the trade to be associated with other related
   trades, e.g. the linkId may contain a tradeId for an associated trade or
   several related trades may be given the same linkId. FpML does not define the
   domain values associated with this element. Note that the domain values for
   this element are not strictly an enumerated list.'

   <allocationTradeId> PartyTradeIdentifier </allocationTradeId> [0..*]
   'The trade id of the allocated trade. This is used by the block trade to
   reference the allocated trade.'

   <blockTradeId> PartyTradeIdentifier </blockTradeId> [0..1]
   'The trade id of the parent trade for N-level allocations. This element is only
   used to model N-level allocations in which the trade acts as block and
   allocated trade at the same time.This basically means the ability to allocate a
   block trade to multiple allocation trades, and then allocate these in turn to
   other allocation trades (and so on if desired).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BlockTradeIdentifier">
   <xsd:complexContent>
      <xsd:extension base=" PartyTradeIdentifier ">
         <xsd:sequence>
            <xsd:element name="allocationTradeId" type=" PartyTradeIdentifier "
            minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="blockTradeId" type=" PartyTradeIdentifier "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ChangeContract

[Table of contents]

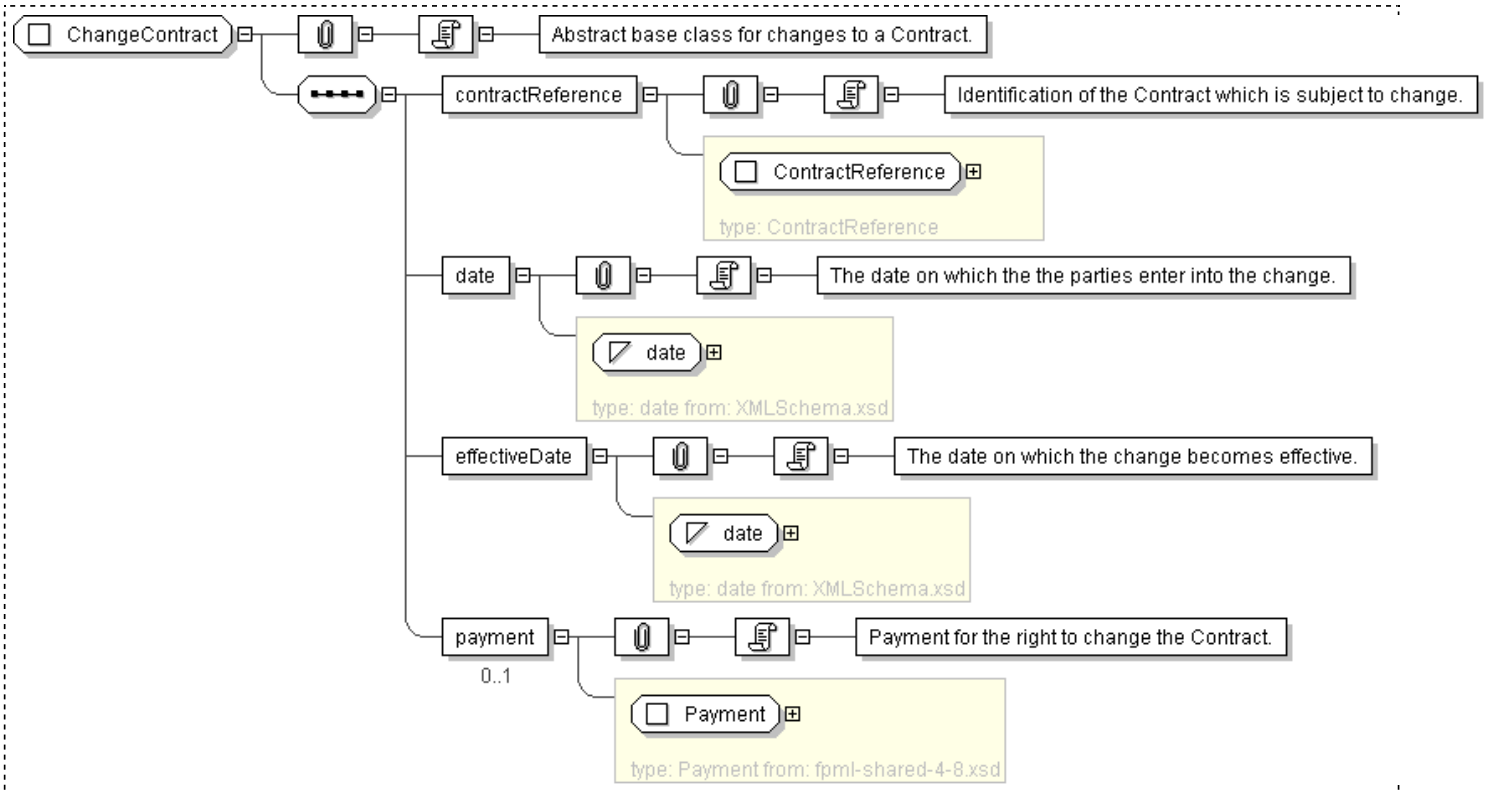
Super-types:	None
Sub-types:	• ChangeContractSize (by extension) • ContractTermination (by extension)

Name	ChangeContract
Abstract	yes
Documentation	Abstract base class for changes to a Contract.

XML Instance Representation

```
<...>
  <contractReference> ContractReference </contractReference> [1]
   'Identification of the Contract which is subject to change.'

  <date> xsd:date </date> [1]
   'The date on which the the parties enter into the change.'

  <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the change becomes effective.'

  <payment> Payment </payment> [0..1]
   'Payment for the right to change the Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ChangeContract" abstract="true">
  <xsd:sequence>
    <xsd:element name="contractReference" type=" ContractReference "/>
    <xsd:element name="date" type=" xsd:date "/>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
```

```
      <xsd:element name="payment" type=" Payment " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ChangeContractSize

[Table of contents]

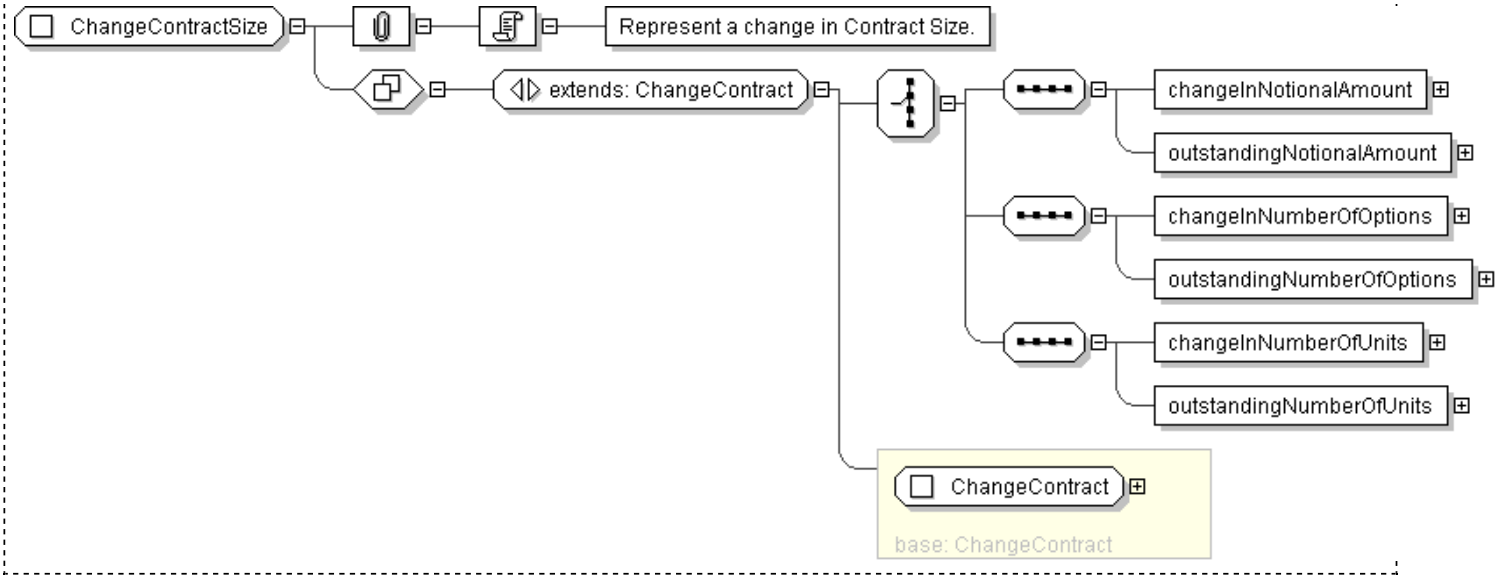
Super-types:	ChangeContract < ChangeContractSize (by extension)
Sub-types:	None

Name	ChangeContractSize
Abstract	no
Documentation	Represent a change in Contract Size.

XML Instance Representation

```
<...>
   <contractReference> ContractReference </contractReference> [1]
   'Identification of the Contract which is subject to change.'

   <date> xsd:date </date> [1]
   'The date on which the the parties enter into the change.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the change becomes effective.'

   <payment> Payment </payment> [0..1]
   'Payment for the right to change the Contract.'

   Start Choice [1]
      <changeInNotionalAmount> Money </changeInNotionalAmount> [1]
      'Specifies the fixed amount by which the Notional Amount changes.'

      <outstandingNotionalAmount> Money </outstandingNotionalAmount> [1]
      'Specifies the Notional amount after the Change.'

      <changeInNumberOfOptions> xsd:decimal </changeInNumberOfOptions> [1]
      'Specifies the fixed amount by which the Number of Options changes.'

      <outstandingNumberOfOptions> xsd:decimal </outstandingNumberOfOptions> [1]
      'Specifies the Number of Options after the Change.'

      <changeInNumberOfUnits> xsd:decimal </changeInNumberOfUnits> [1]
      'Specifies the fixed amount by which the Number of Units changes.'

      <outstandingNumberOfUnits> xsd:decimal </outstandingNumberOfUnits> [1]
      'Specifies the Number of Units.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ChangeContractSize">
  <xsd:complexContent>
    <xsd:extension base=" ChangeContract ">
      <xsd:choice>
        <xsd:sequence>
          <xsd:element name="changeInNotionalAmount" type=" Money "/>
          <xsd:element name="outstandingNotionalAmount" type=" Money "/>
        </xsd:sequence>
        <xsd:sequence>
          <xsd:element name="changeInNumberOfOptions" type=" xsd:decimal "/>
          <xsd:element name="outstandingNumberOfOptions" type=" xsd:decimal "/>
        </xsd:sequence>
        <xsd:sequence>
          <xsd:element name="changeInNumberOfUnits" type=" xsd:decimal "/>
          <xsd:element name="outstandingNumberOfUnits" type=" xsd:decimal "/>
        </xsd:sequence>
      </xsd:choice>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Collateral

[Table of contents]

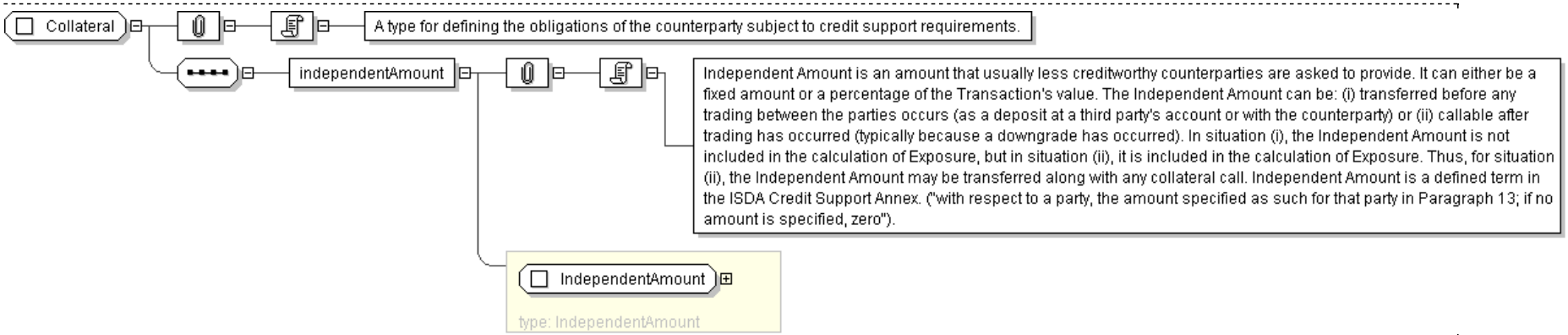
Super-types:	None
Sub-types:	None

Name	Collateral
Used by (from the same schema document)	Complex Type Contract , Complex Type Trade , Model Group AllocationContent.model
Abstract	no
Documentation	A type for defining the obligations of the counterparty subject to credit support requirements.

XML Instance Representation

```
<...>
  <independentAmount> IndependentAmount </independentAmount> [1]
  'Independent Amount is an amount that usually less creditworthy counterparties are asked to provide. It can either be a fixed amount
  or a percentage of the Transaction\'s value. The Independent Amount can be: (i) transferred before any trading between the parties
  occurs (as a deposit at a third party\'s account or with the counterparty) or (ii) callable after trading has occurred (typically
  because a downgrade has occurred). In situation (i), the Independent Amount is not included in the calculation of Exposure, but in
  situation (ii), it is included in the calculation of Exposure. Thus, for situation (ii), the Independent Amount may be transferred
  along with any collateral call. Independent Amount is a defined term in the ISDA Credit Support Annex. (\"with respect to a party, the
  amount specified as such for that party in Paragraph 13; if no amount is specified, zero\").'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Collateral">
  <xsd:sequence>
    <xsd:element name="independentAmount" type=" IndependentAmount "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Contract

[Table of contents]

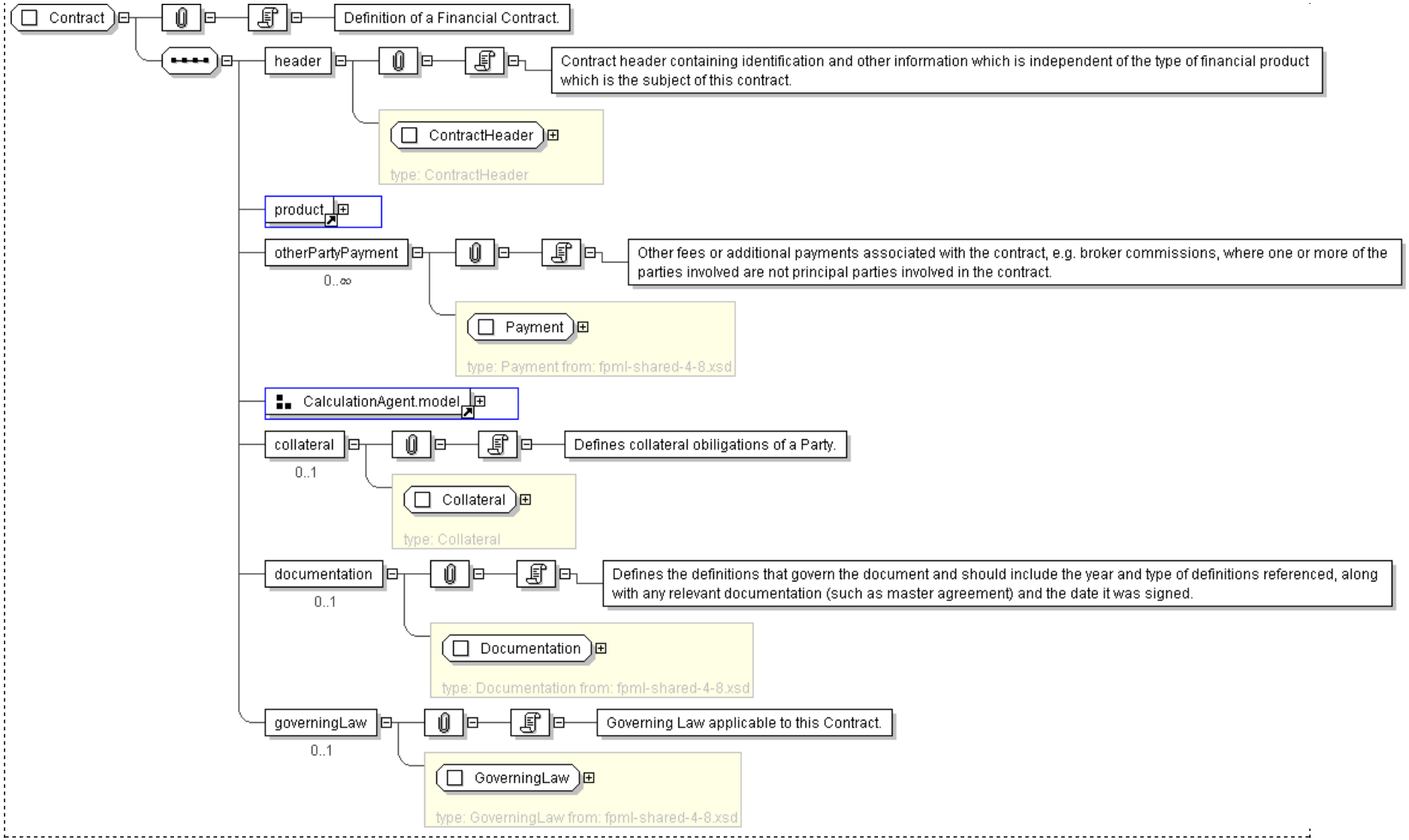
Super-types:	None
Sub-types:	None

Name	Contract
Used by (from the same schema document)	Model Group ContractNovationDetails.model , Model Group ContractNovationDetails.model , Model Group ContractNovationDetails.model , Model Group ContractOrContractReference.model
Abstract	no
Documentation	Definition of a Financial Contract.

XML Instance Representation

```
<...>
   <header> ContractHeader </header> [1]
   'Contract header containing identification and other information which is independent of the type of financial product which is
   the subject of this contract.'

   <product> ... </product> [1]
   <otherPartyPayment> Payment </otherPartyPayment> [0..*]
   'Other fees or additional payments associated with the contract, e.g. broker commissions, where one or more of the parties
   involved are not principal parties involved in the contract.'

   <calculationAgent> CalculationAgent </calculationAgent> [0..1]
   'The ISDA calculation agent responsible for performing duties as defined in the applicable product definitions.'

   <calculationAgentBusinessCenter> BusinessCenter </calculationAgentBusinessCenter> [0..1]
   'The city in which the office through which ISDA Calculation Agent is acting for purposes of the transaction is located The
   short-form confirm for a trade that is executed under a Sovereign or Asia Pacific Master Confirmation Agreement ( MCA ), does not
   need to specify the Calculation Agent. However, the confirm does need to specify the Calculation Agent City. This is due to the
   fact that the MCA sets the value for Calculation Agent but does not set the value for Calculation Agent City.'

   <collateral> Collateral </collateral> [0..1]
   'Defines collateral obiligations of a Party.'

   <documentation> Documentation </documentation> [0..1]
   'Defines the definitions that govern the document and should include the year and type of definitions referenced, along with any
   relevant documentation (such as master agreement) and the date it was signed.'

   <governingLaw> GoverningLaw </governingLaw> [0..1]
   'Governing Law applicable to this Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Contract">
  <xsd:sequence>
    <xsd:element name="header" type=" ContractHeader "/>
    <xsd:element ref=" product "/>
    <xsd:element name="otherPartyPayment" type=" Payment " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:group ref=" CalculationAgent.model "/>
    <xsd:element name="collateral" type=" Collateral " minOccurs="0"/>
    <xsd:element name="documentation" type=" Documentation " minOccurs="0"/>
    <xsd:element name="governingLaw" type=" GoverningLaw " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractHeader

[Table of contents]

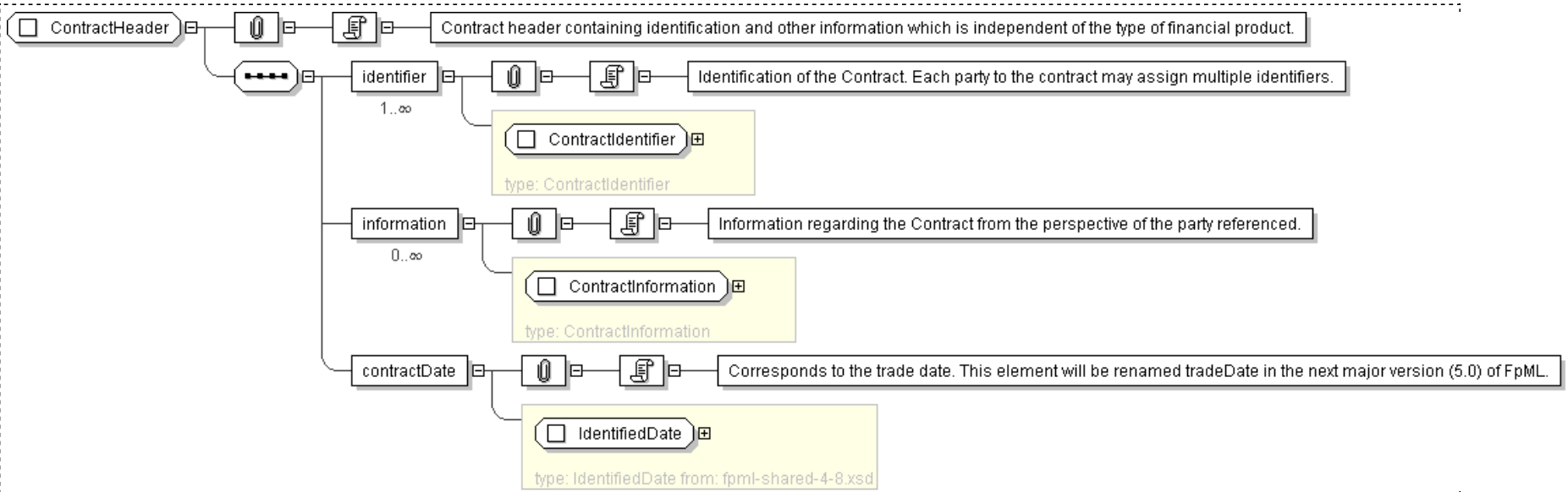
Super-types:	None
Sub-types:	None

Name	ContractHeader
Used by (from the same schema document)	Complex Type Contract
Abstract	no
Documentation	Contract header containing identification and other information which is independent of the type of financial product.

XML Instance Representation

```
<...>
  <identifier> ContractIdentifier </identifier> [1..*]
  'Identification of the Contract. Each party to the contract may assign multiple identifiers.'

  <information> ContractInformation </information> [0..*]
  'Information regarding the Contract from the perspective of the party referenced.'

  <contractDate> IdentifiedDate </contractDate> [1]
  'Corresponds to the trade date. This element will be renamed tradeDate in the next major version (5.0) of FpML.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractHeader">
   <xsd:sequence>
      <xsd:element name="identifier" type=" ContractIdentifier " maxOccurs="unbounded"/>
      <xsd:element name="information" type=" ContractInformation " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="contractDate" type=" IdentifiedDate "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractId

[Table of contents]

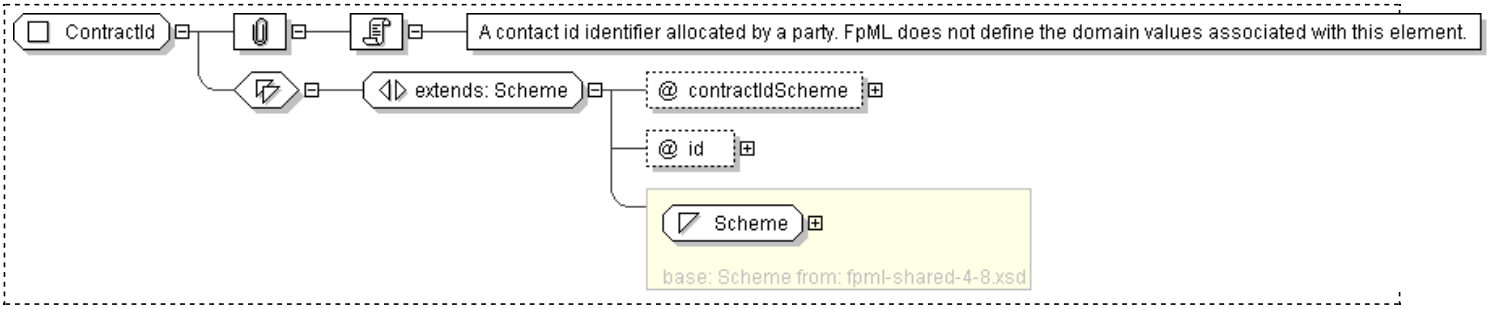
Super-types:	Scheme < ContractId (by extension)
Sub-types:	None

Name	ContractId
Used by (from the same schema document)	Complex Type ContractIdentifier , Complex Type VersionedContractId
Abstract	no
Documentation	A contact id identifier allocated by a party. FpML does not define the domain values associated with this element.

XML Instance Representation

```
<...
 contractIdScheme=" xsd:anyURI [1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="contractIdScheme" type=" xsd:anyURI " use="required"/>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractIdentifier

[Table of contents]

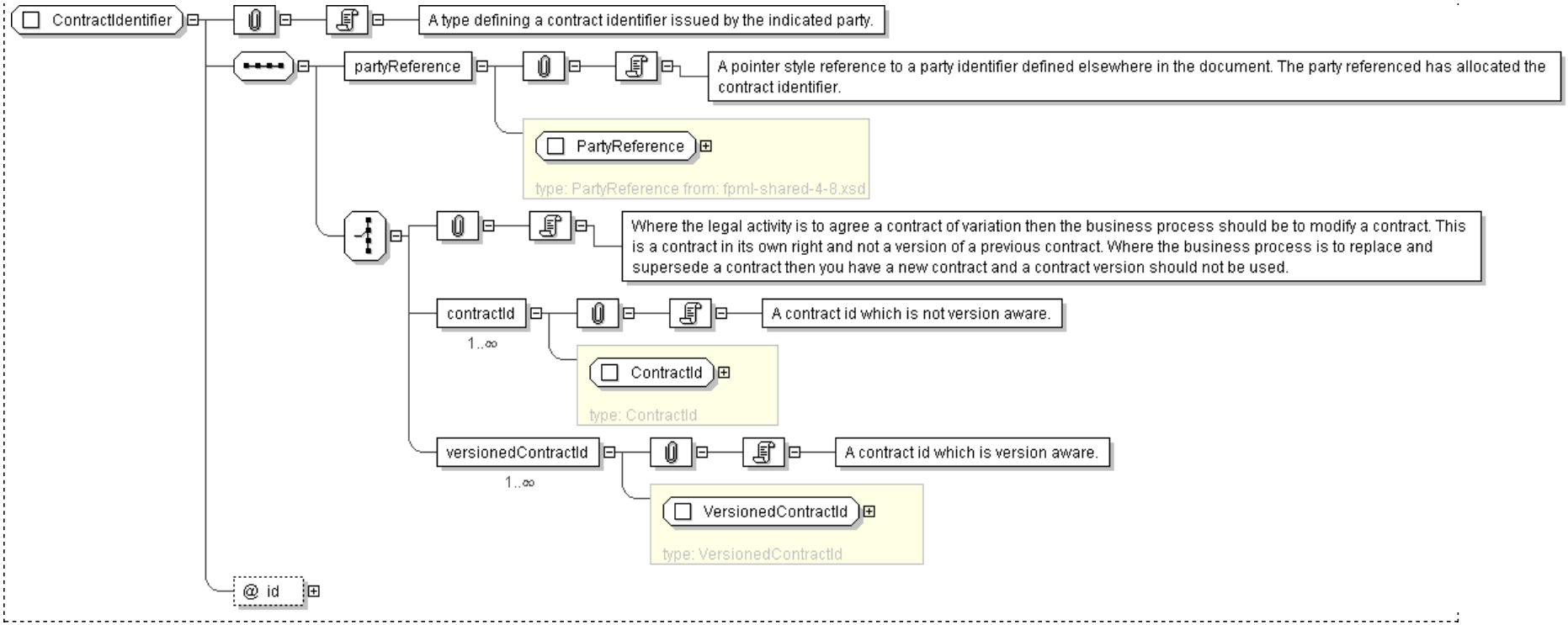
Super-types:	None
Sub-types:	None

Name	ContractIdentifier
Used by (from the same schema document)	Complex Type ContractHeader , Complex Type ContractReference
Abstract	no
Documentation	A type defining a contract identifier issued by the indicated party.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
    'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced has allocated the contract
    identifier.'

   Start Choice [1]
'Where the legal activity is to agree a contract of variation then the business process should be to modify a contract. This is a
contract in its own right and not a version of a previous contract. Where the business process is to replace and supersede a contract
then you have a new contract and a contract version should not be used.'


      <contractId> ContractId </contractId> [1..*]
       'A contract id which is not version aware.'

      <versionedContractId> VersionedContractId </versionedContractId> [1..*]
       'A contract id which is version aware.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractIdentifier">
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
    <xsd:choice>
      <xsd:element name="contractId" type=" ContractId " maxOccurs="unbounded"/>
      <xsd:element name="versionedContractId" type=" VersionedContractId " maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractInformation

[Table of contents]

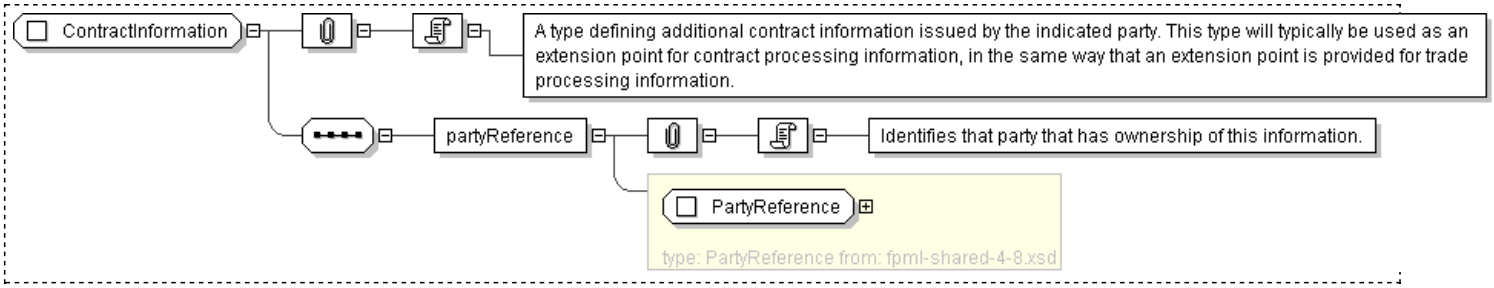
Super-types:	None
Sub-types:	None

Name	ContractInformation
Used by (from the same schema document)	Complex Type ContractHeader
Abstract	no
Documentation	A type defining additional contract information issued by the indicated party. This type will typically be used as an extension point for contract processing information, in the same way that an extension point is provided for trade processing information.

XML Instance Representation

```
<...>
  <partyReference> PartyReference </partyReference> [1]
   'Identifies that party that has ownership of this information.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractInformation">
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractNovation

[Table of contents]

Super-types:	None
Sub-types:	None

Name	ContractNovation
Abstract	no
Documentation	Details of the Contract Novation.

XML Instance Representation

```
<...>
  Start Choice [1]
    Start Choice [1]
'Choice between identification and representation of the new contract.'


      <newContractReference> ContractReference </newContractReference> [1]
      'Indicates a reference to the new Contract between the transferee and the
      remaining party.'

      <newContract> Contract </newContract> [1]
      'Indicates the new Contract between the transferee and the remaining
      party.'

    End Choice
    Start Choice [1]
      <oldContractReference> ContractReference </oldContractReference> [1]
      'Indicates a reference to the original contract between the transferor and
      the remaining party.'

      <oldContract> Contract </oldContract> [1]
      'Indicates the original Contract between the transferor and the remaining
      party.'

    End Choice
    Start Choice [0..1]
'Choice between identification and representation of the new contract.'


      <newContractReference> ContractReference </newContractReference> [1]
      <newContract> Contract </newContract> [1]
    End Choice
  End Choice
  <transferor> PartyReference </transferor> [1]
  'A pointer style reference to a party identifier defined elsewhere in the
  document. In a three-way novation the party referenced is the Transferor
  (outgoing party) in the novation. The Transferor means a party which transfers
  by novation to a Transferee all of its rights, liabilities, duties and
  obligations with respect to a Remaining Party. In a four-way novation the party
  referenced is Transferor 1 which transfers by novation to Transferee 1 all of
  its rights, liabilities, duties and obligations with respect to Transferor 2.
  ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-
  way novation).'

  <transferee> PartyReference </transferee> [1]
  'A pointer style reference to a party identifier defined elsewhere in the
  document. In a three-way novation the party referenced is the Transferee
  (incoming party) in the novation. Transferee means a party which accepts by way
  of novation all rights, liabilities, duties and obligations of a Transferor
  with respect to a Remaining Party. In a four-way novation the party referenced
  is Transferee 1 which accepts by way of novation the rights, liabilities,
```

duties and obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).'

<remainingParty> PartyReference </remainingParty> [1]

'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the novation. Remaining Party means a party which consents to a Transferor\'s transfer by novation and the acceptance thereof by the Transferee of all of the Transferor\'s rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation) or Transferor 2 (four-way novation).'

<otherRemainingParty> PartyReference </otherRemainingParty> [0..1]

'A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).'

<novationDate> xsd:date </novationDate> [1]

'Specifies the date that one party\'s legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of the 2004 ISDA Novation Definitions, section 1.16.'

<novationContractDate> xsd:date </novationContractDate> [0..1]

'Specifies the date the parties agree to assign or novate a Contract. If this element is not specified, the novationContractDate will be deemed to be the novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.'

Start Choice [1]

'Choice for expressing the novated amount as either a money amount, number of options, or number of units, according the the financial product which is being novated.'

<novatedAmount> Money </novatedAmount> [1]

'The amount which represents the portion of the Old Contract being novated.'

<novatedNumberOfOptions> xsd:decimal </novatedNumberOfOptions> [1]

'The number of options which represent the portion of the Old Contract being novated.'

<novatedNumberOfUnits> xsd:decimal </novatedNumberOfUnits> [1]

'The number of options which represent the portion of the Old Contract being novated.'

End Choice

<fullFirstCalculationPeriod> xsd:boolean </fullFirstCalculationPeriod> [0..1]

'This element corresponds to the applicability of the Full First Calculation Period as defined in the 2004 ISDA Novation Definitions, section 1.20.'

<firstPeriodStartDate> FirstPeriodStartDate </firstPeriodStartDate> [0..2]

'Element that is used to be able to make sense of the â€œnew transactionâ€ without requiring reference back to the â€œold transactionâ€. In the case of interest rate products there are potentially 2 â€œfirst period start datesâ€ to reference â€“ one with respect to each party to the new transaction. For Credit Default Swaps there is just the one with respect to the party that is the fixed rate payer.'

<nonReliance> Empty </nonReliance> [0..1]

'This element corresponds to the non-Reliance section in the 2004 ISDA Novation Definitions, section 2.1 (c) (i). The element appears in the instance document when non-Reliance is applicable.'

<creditDerivativesNotices> CreditDerivativesNotices </creditDerivativesNotices> [0..1]

'This element should be specified if one or more of either a Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party. The type of notice or notices that

have been delivered should be indicated by setting the relevant boolean element value(s) to true. The absence of the element means that no Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party.'

<contractualDefinitions> ContractualDefinitions </contractualDefinitions> [0..*]

'The definitions (such as those published by ISDA) that will define the terms of the novation transaction.'

<contractualTermsSupplement> ContractualTermsSupplement </contractualTermsSupplement> [0..*]

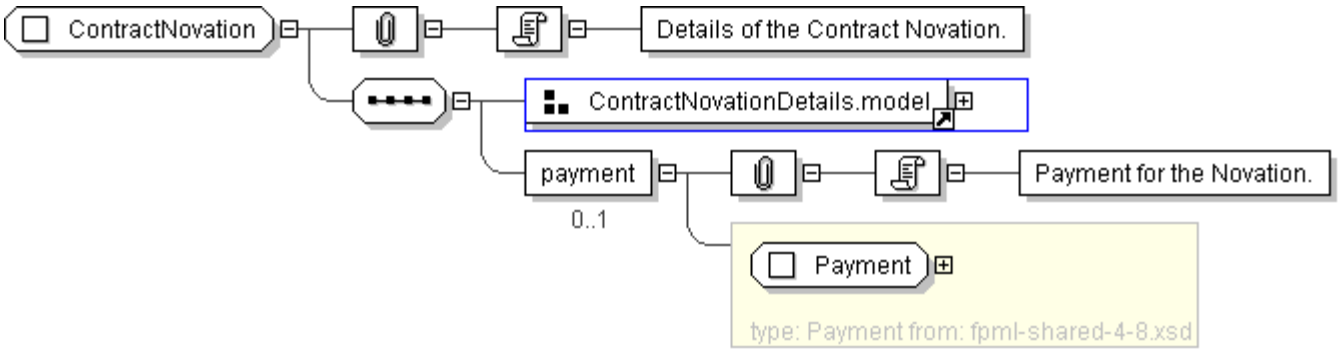
'A contractual supplement (such as those published by ISDA) that will apply to the trade.'

<payment> Payment </payment> [0..1]

'Payment for the Novation.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractNovation">
  <xsd:sequence>
    <xsd:group ref=" ContractNovationDetails.model "/>
    <xsd:element name="payment" type=" Payment " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractReference

[Table of contents]

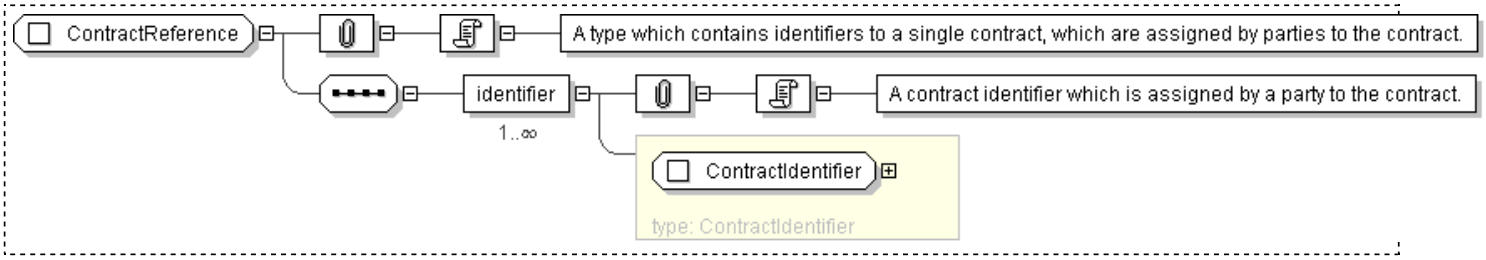
Super-types:	None
Sub-types:	None

Name	ContractReference
Used by (from the same schema document)	Complex Type ChangeContract , Model Group ContractNovationDetails.model , Model Group ContractNovationDetails.model , Model Group ContractNovationDetails.model , Model Group ContractOrContractReference.model
Abstract	no
Documentation	A type which contains identifiers to a single contract, which are assigned by parties to the contract.

XML Instance Representation

```
<...>
  <identifier> ContractIdentifier </identifier> [1..*]
   'A contract identifier which is assigned by a party to the contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractReference">
  <xsd:sequence>
    <xsd:element name="identifier" type=" ContractIdentifier " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractTermination

[Table of contents]

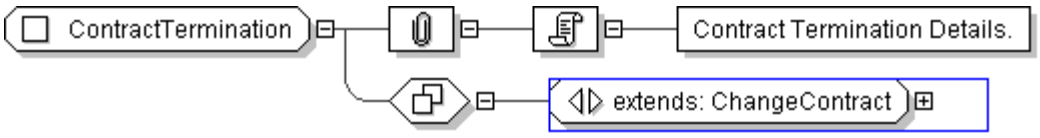
Super-types:	ChangeContract < ContractTermination (by extension)
Sub-types:	None

Name	ContractTermination
Abstract	no
Documentation	Contract Termination Details.

XML Instance Representation

```
<...>
   <contractReference> ContractReference </contractReference> [1]
   'Identification of the Contract which is subject to change.'

   <date> xsd:date </date> [1]
   'The date on which the the parties enter into the change.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the change becomes effective.'

   <payment> Payment </payment> [0..1]
   'Payment for the right to change the Contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractTermination">
   <xsd:complexContent>
      <xsd:extension base=" ChangeContract "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditDerivativesNotices

[Table of contents]

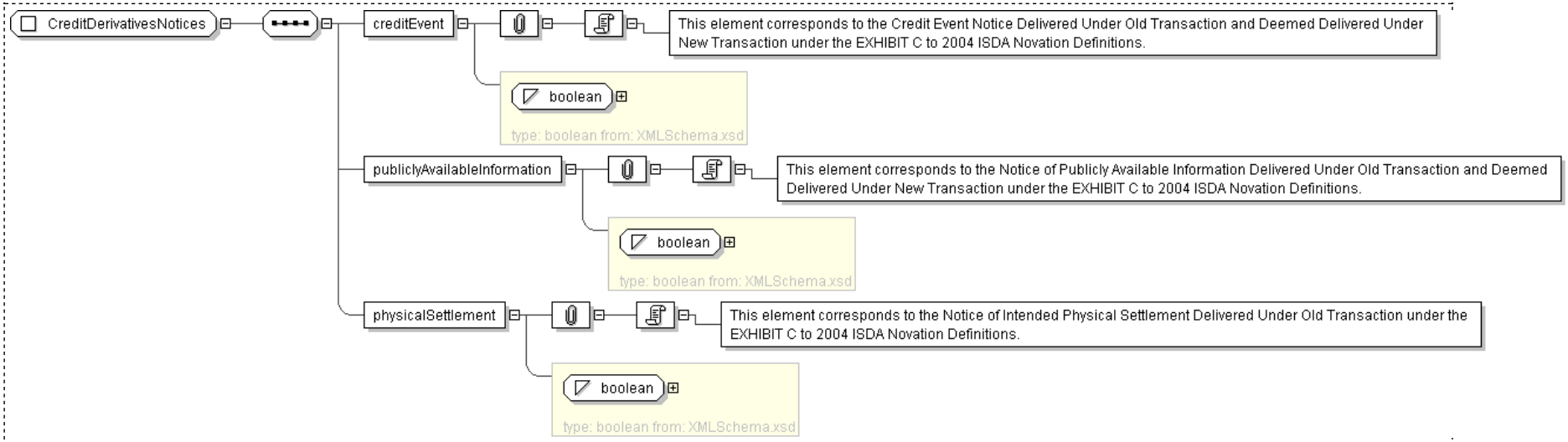
Super-types:	None
Sub-types:	None

Name	CreditDerivativesNotices
Used by (from the same schema document)	Model Group ContractNovationDetails.model
Abstract	no

XML Instance Representation

```
<...>
   <creditEvent> xsd:boolean </creditEvent> [1]
   'This element corresponds to the Credit Event Notice Delivered Under Old Transaction and Deemed Delivered Under New Transaction under the EXHIBIT C
   to 2004 ISDA Novation Definitions.'

   <publiclyAvailableInformation> xsd:boolean </publiclyAvailableInformation> [1]
   'This element corresponds to the Notice of Publicly Available Information Delivered Under Old Transaction and Deemed Delivered Under New Transaction
   under the EXHIBIT C to 2004 ISDA Novation Definitions.'

   <physicalSettlement> xsd:boolean </physicalSettlement> [1]
   'This element corresponds to the Notice of Intended Physical Settlement Delivered Under Old Transaction under the EXHIBIT C to 2004 ISDA Novation
   Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditDerivativesNotices">
   <xsd:sequence>
      <xsd:element name="creditEvent" type=" xsd:boolean "/>
      <xsd:element name="publiclyAvailableInformation" type=" xsd:boolean "/>
      <xsd:element name="physicalSettlement" type=" xsd:boolean "/>
   </xsd:sequence>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: DataDocument

[Table of contents]

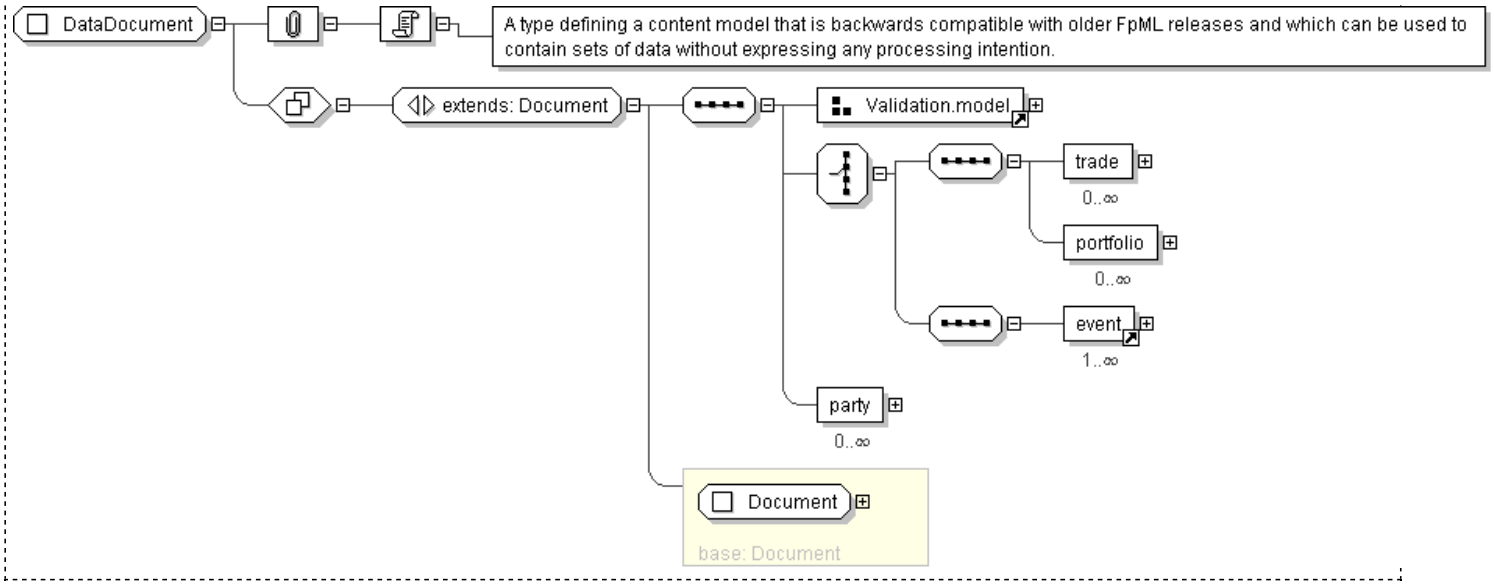
Super-types:	Document < DataDocument (by extension)
Sub-types:	None

Name	DataDocument
Abstract	no
Documentation	A type defining a content model that is backwards compatible with older FpML releases and which can be used to contain sets of data without expressing any processing intention.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
 changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
 the actual build number stays the same.'

 ">
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [0..*]
      'The root element in an FpML trade document.'

      <portfolio> Portfolio </portfolio> [0..*]
      'An arbitary grouping of trade references (and possibly other portfolios).'

      <event> ... </event> [1..*]
      'A business event.'

   End Choice
   <party> Party </party> [0..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the
   term of the trade, but may include other parties involved in, or incidental to, the trade, such as
   parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML
   roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DataDocument">
  <xsd:complexContent>
    <xsd:extension base=" Document ">
      <xsd:sequence>
        <xsd:group ref=" Validation.model "/>
        <xsd:choice>
          <xsd:sequence>
            <xsd:element name="trade" type=" Trade " minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="portfolio" type=" Portfolio " minOccurs="0" maxOccurs="unbounded"/>
          </xsd:sequence>
          <xsd:sequence>
            <xsd:element ref=" event " maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:choice>
        <xsd:element name="party" type=" Party " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Document

[Table of contents]

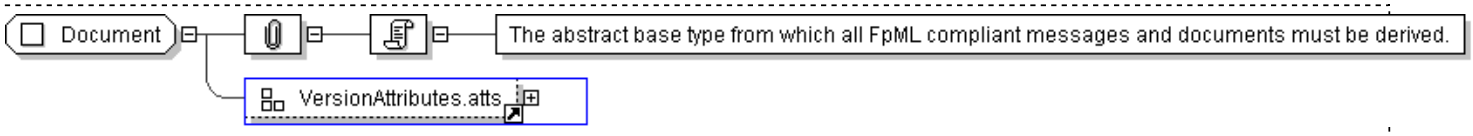
Super-types:	None
Sub-types:	• DataDocument (by extension)

Name	Document
Abstract	yes
Documentation	The abstract base type from which all FpML compliant messages and documents must be derived.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 "/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Document" abstract="true">
   <xsd:attributeGroup ref=" VersionAttributes.atts "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Event

[Table of contents]

Super-types: None

Sub-types:

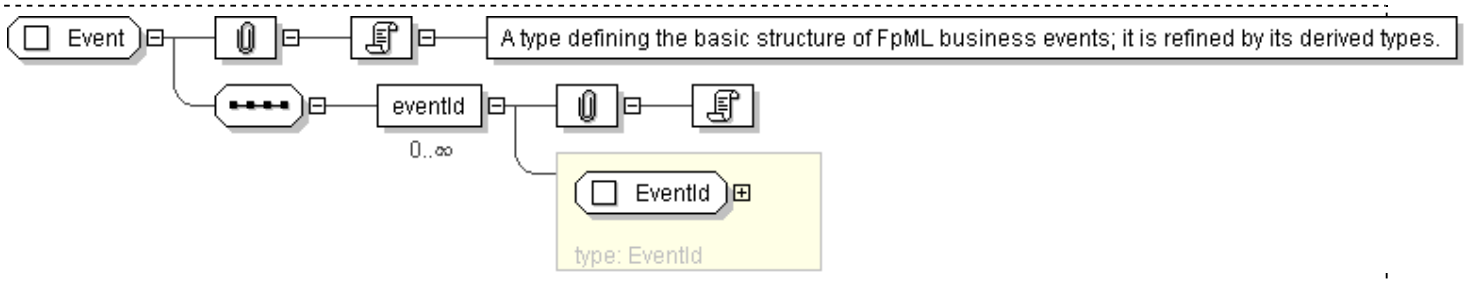
- Amendment (by extension)
- Increase (by extension)

Name	Event
Used by (from the same schema document)	Element event
Abstract	yes
Documentation	A type defining the basic structure of FpML business events; it is refined by its derived types.

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Event" abstract="true">
   <xsd:sequence>
      <xsd:element name="eventId" type=" EventId " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EventId

[Table of contents]

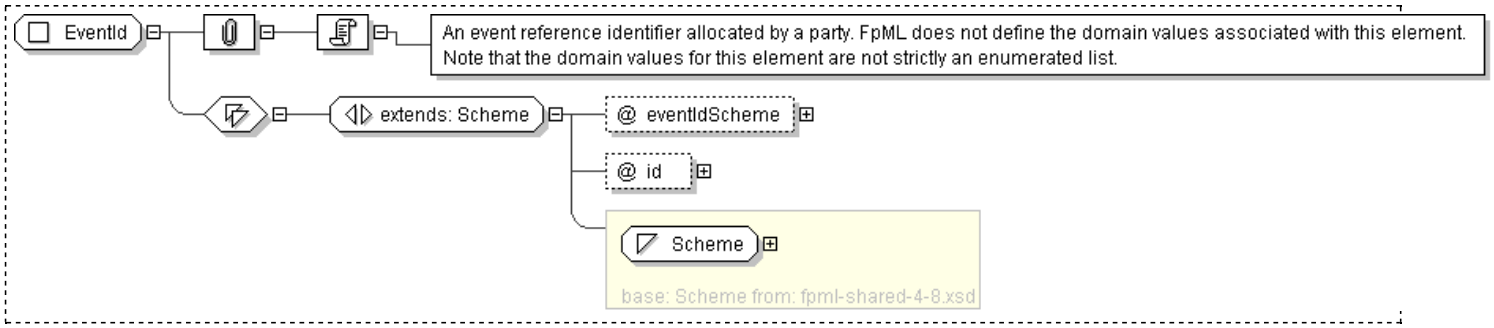
Super-types:	Scheme < EventId (by extension)
Sub-types:	None

Name	EventId
Used by (from the same schema document)	Complex Type Event
Abstract	no
Documentation	An event reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not strictly an enumerated list.

XML Instance Representation

```
<...
 eventIdScheme=" xsd:anyURI [1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EventId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="eventIdScheme" type=" xsd:anyURI " use="required"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExecutionDateTime

[Table of contents]

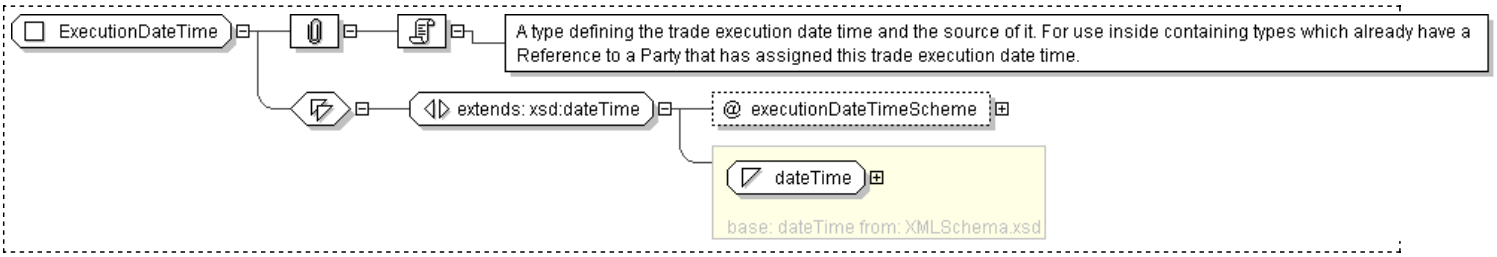
Super-types:	xsd:dateTime < ExecutionDateTime (by extension)
Sub-types:	None

Name	ExecutionDateTime
Used by (from the same schema document)	Complex Type PartyTradeInformation
Abstract	no
Documentation	A type defining the trade execution date time and the source of it. For use inside containing types which already have a Reference to a Party that has assigned this trade execution date time.

XML Instance Representation

```
<...
 executionDateTimeScheme=" xsd:anyURI [0..1]
 'Identification of the source (e.g. clock id) generating the execution date time.'

 ">
   xsd:dateTime
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExecutionDateTime">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:dateTime ">
      <xsd:attribute name="executionDateTimeScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FirstPeriodStartDate

[Table of contents]

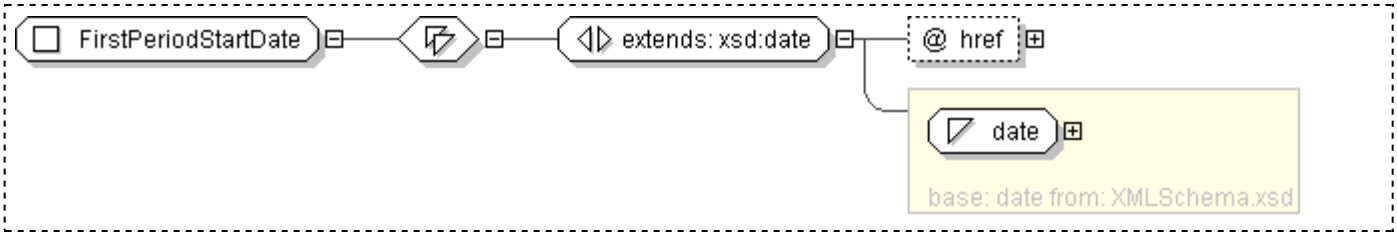
Super-types:	xsd:date < FirstPeriodStartDate (by extension)
Sub-types:	None

Name	FirstPeriodStartDate
Used by (from the same schema document)	Model Group ContractNovationDetails.model
Abstract	no

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]">
   xsd:date
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FirstPeriodStartDate">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:date ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="Party"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Increase

[Table of contents]

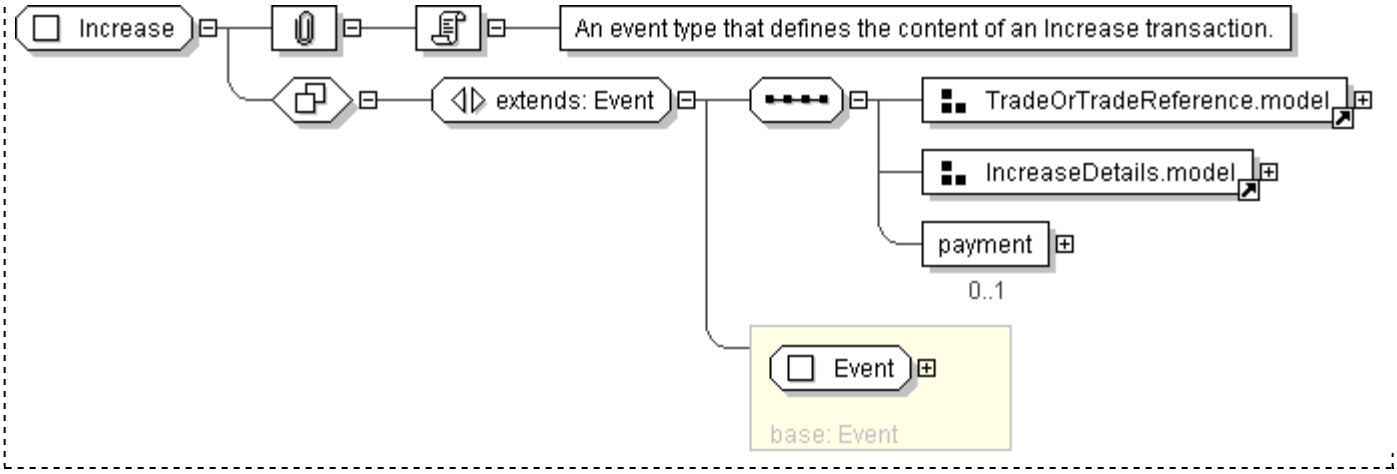
Super-types:	Event < Increase (by extension)
Sub-types:	None

Name	Increase
Abstract	no
Documentation	An event type that defines the content of an Increase transaction.

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a
      mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more
      different partyTradeIdentifier elements - each allocated by a different
      party.'

   End Choice
   <increaseTradeDate> xsd:date </increaseTradeDate> [1]
   'The date on which the the parties enter into the Increase transaction'

   <increaseEffectiveDate> xsd:date </increaseEffectiveDate> [1]
   'The date on which the Increase becomes effective'

   Start Choice [1]
      <increaseInNotionalAmount> Money </increaseInNotionalAmount> [1]
      'Specifies the fixed amount by which the Notional increases due to the
      Increase transaction.'

      <outstandingNotionalAmount> Money </outstandingNotionalAmount> [1]
      'Specifies the Notional amount after the Increase.'

      <increaseInNumberOfOptions> xsd:decimal </increaseInNumberOfOptions> [1]
      'Specifies the fixed amount by which the Number of Options increases due to
      the Increase transaction.'

      <outstandingNumberOfOptions> xsd:decimal </outstandingNumberOfOptions> [1]
      'Specifies the Number of Options after the Increase.'

   End Choice
   <payment> Payment </payment> [0..1]
   'A payment for the right to increase the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Increase">
   <xsd:complexContent>
      <xsd:extension base=" Event ">
         <xsd:sequence>
            <xsd:group ref=" TradeOrTradeReference.model "/>
            <xsd:group ref=" IncreaseDetails.model "/>
            <xsd:element name="payment" type=" Payment " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndependentAmount

[Table of contents]

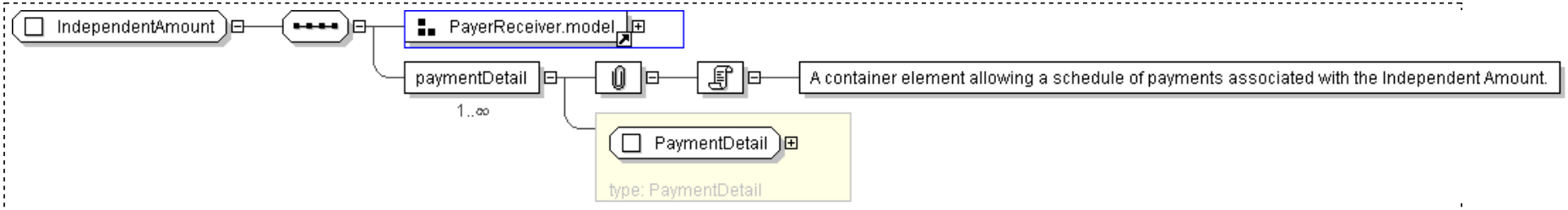
Super-types:	None
Sub-types:	None

Name	IndependentAmount
Used by (from the same schema document)	Complex Type Collateral
Abstract	no

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

   <paymentDetail> PaymentDetail </paymentDetail> [1..*]
    'A container element allowing a schedule of payments associated with the Independent Amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndependentAmount">
   <xsd:sequence>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="paymentDetail" type=" PaymentDetail " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LinkId

[Table of contents]

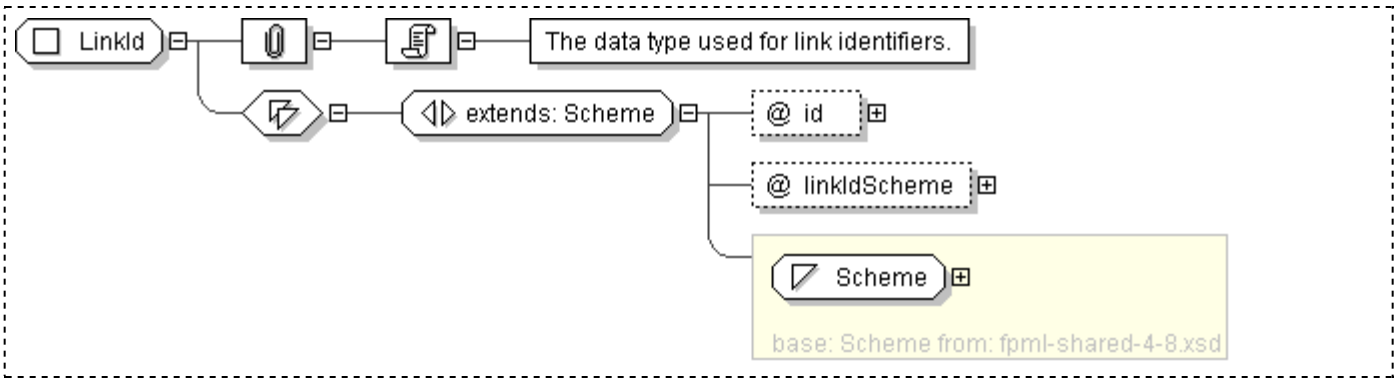
Super-types:	Scheme < LinkId (by extension)
Sub-types:	None

Name	LinkId
Used by (from the same schema document)	Complex Type PartyTradeIdentifier
Abstract	no
Documentation	The data type used for link identifiers.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 linkIdScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LinkId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="id" type=" xsd:ID "/>
         <xsd:attribute name="linkIdScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyPortfolioName

[Table of contents]

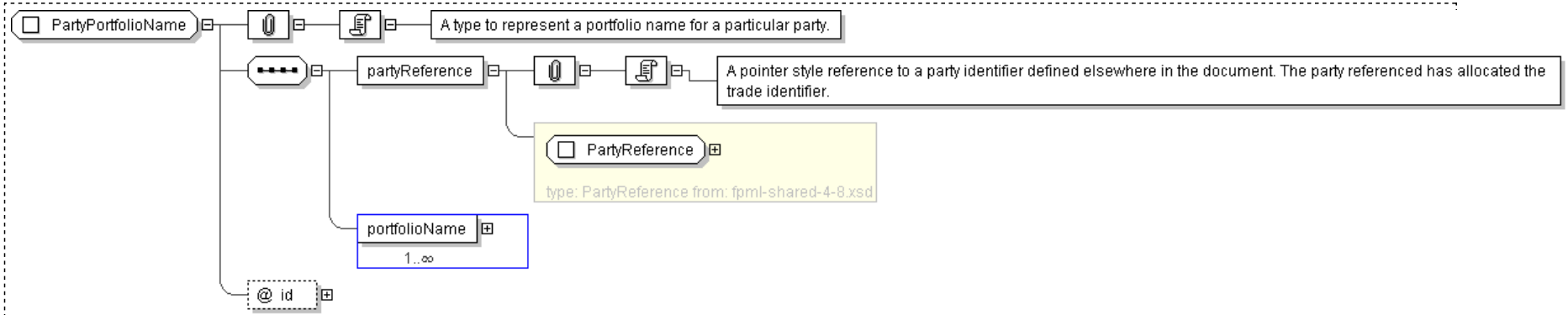
Super-types:	None
Sub-types:	None

Name	PartyPortfolioName
Used by (from the same schema document)	Complex Type Portfolio
Abstract	no
Documentation	A type to represent a portfolio name for a particular party.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
    'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced has allocated the trade
    identifier.'

   <portfolioName> PortfolioName </portfolioName> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyPortfolioName">
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
    <xsd:element name="portfolioName" type=" PortfolioName " maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyRole

[Table of contents]

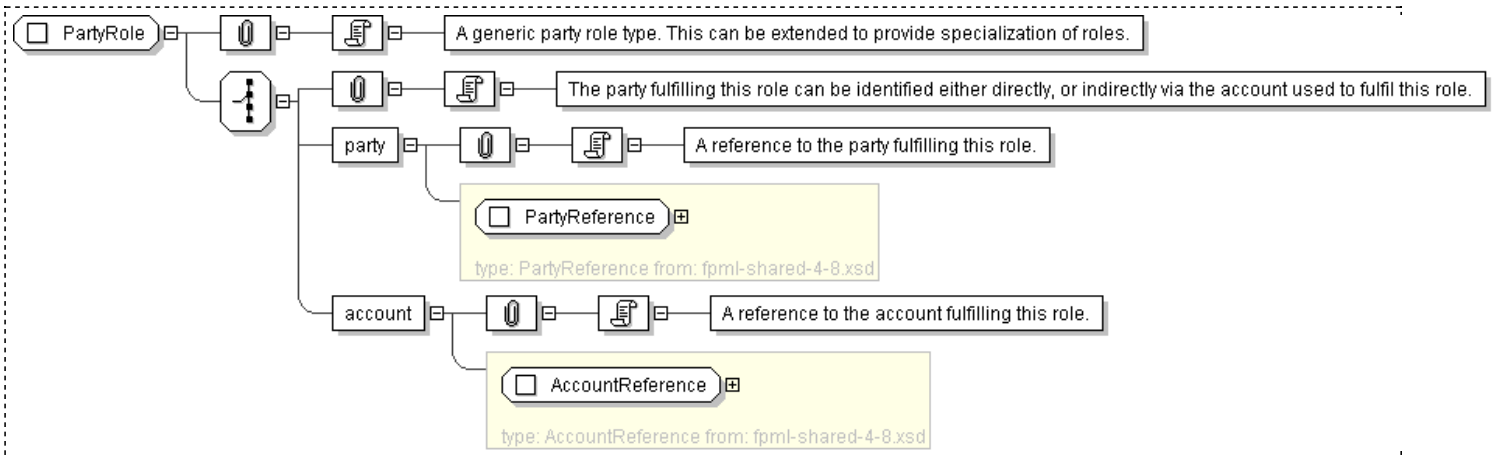
Super-types:	None
Sub-types:	None

Name	PartyRole
Used by (from the same schema document)	Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide , Complex Type TradeSide
Abstract	no
Documentation	A generic party role type. This can be extended to provide specialization of roles.

XML Instance Representation

```
<...>
  Start Choice [1]
'The party fulfilling this role can be identified either directly, or indirectly via the account used to
fulfil this role.'


     <party> PartyReference </party> [1]
     'A reference to the party fulfilling this role.'

     <account> AccountReference </account> [1]
     'A reference to the account fulfilling this role.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyRole">
  <xsd:choice>
    <xsd:element name="party" type=" PartyReference "/>
    <xsd:element name="account" type=" AccountReference "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyTradeIdentifier

[Table of contents]

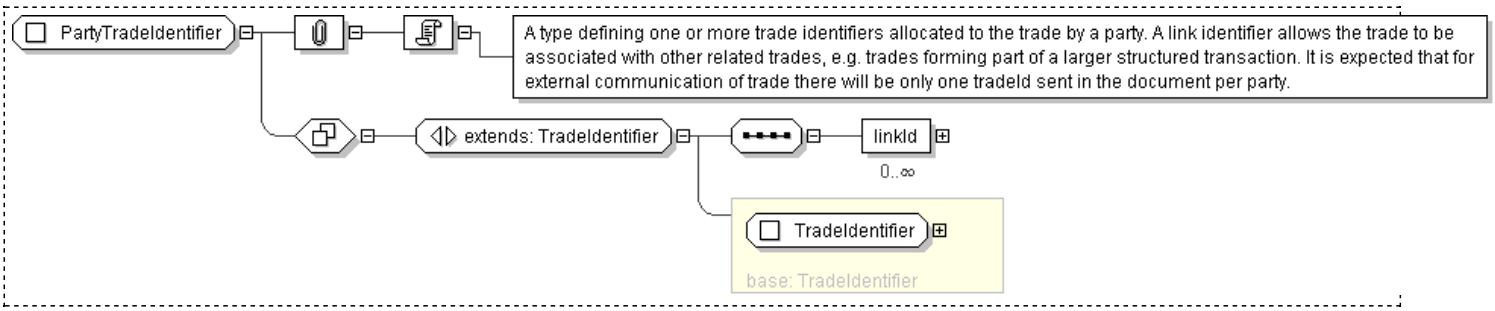
Super-types:	TradeIdentifier < PartyTradeIdentifier (by extension)
Sub-types:	• AllocationTradeIdentifier (by extension) • BlockTradeIdentifier (by extension)

Name	PartyTradeIdentifier
Used by (from the same schema document)	Complex Type Allocation , Complex Type AllocationTradeIdentifier , Complex Type BlockTradeIdentifier , Complex Type BlockTradeIdentifier , Complex Type PartyTradeIdentifiers , Complex Type TradeHeader
Abstract	no
Documentation	A type defining one or more trade identifiers allocated to the trade by a party. A link identifier allows the trade to be associated with other related trades, e.g. trades forming part of a larger structured transaction. It is expected that for external communication of trade there will be only one tradeId sent in the document per party.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced has
   allocated the trade identifier.'

   Start Choice [1..*]
      <tradeId> TradeId </tradeId> [1]
      <versionedTradeId> VersionedTradeId </versionedTradeId> [1]
   End Choice
   <linkId> LinkId </linkId> [0..*]
   'A link identifier allowing the trade to be associated with other related trades, e.g. the linkId may
   contain a tradeId for an associated trade or several related trades may be given the same linkId. FpML does
   not define the domain values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyTradeIdentifier">
  <xsd:complexContent>
    <xsd:extension base=" TradeIdentifier ">
      <xsd:sequence>
        <xsd:element name="linkId" type=" LinkId " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyTradeIdentifiers

[Table of contents]

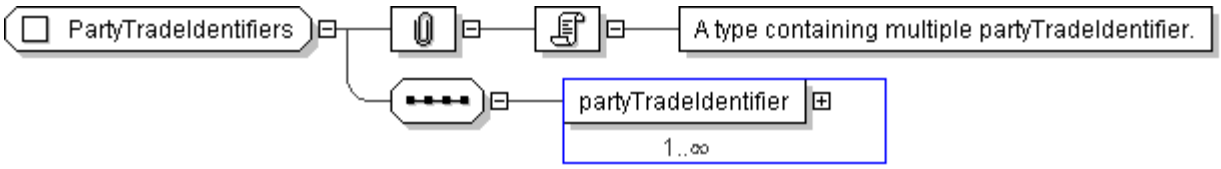
Super-types:	None
Sub-types:	None

Name	PartyTradeIdentifiers
Used by (from the same schema document)	Model Group TradeOrTradeReference.model
Abstract	no
Documentation	A type containing multiple partyTradeIdentifier.

XML Instance Representation

```
<...>
   <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyTradeIdentifiers">
   <xsd:sequence>
      <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier "
      maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyTradeInformation

[Table of contents]

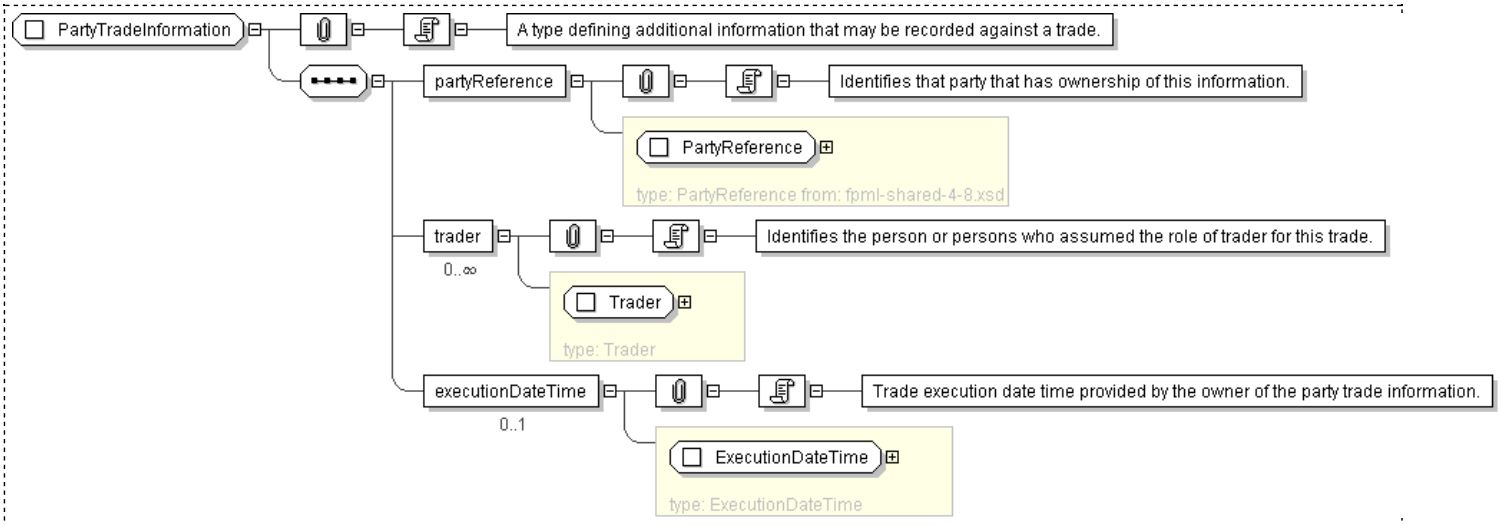
Super-types:	None
Sub-types:	None

Name	PartyTradeInformation
Used by (from the same schema document)	Complex Type TradeHeader
Abstract	no
Documentation	A type defining additional information that may be recorded against a trade.

XML Instance Representation

```
<...>
  <partyReference> PartyReference </partyReference> [1]
  'Identifies that party that has ownership of this information.'

  <trader> Trader </trader> [0..*]
  'Identifies the person or persons who assumed the role of trader for this trade.'

  <executionDateTime> ExecutionDateTime </executionDateTime> [0..1]
  'Trade execution date time provided by the owner of the party trade information.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyTradeInformation">
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
    <xsd:element name="trader" type=" Trader " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="executionDateTime" type=" ExecutionDateTime " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentDetail

[Table of contents]

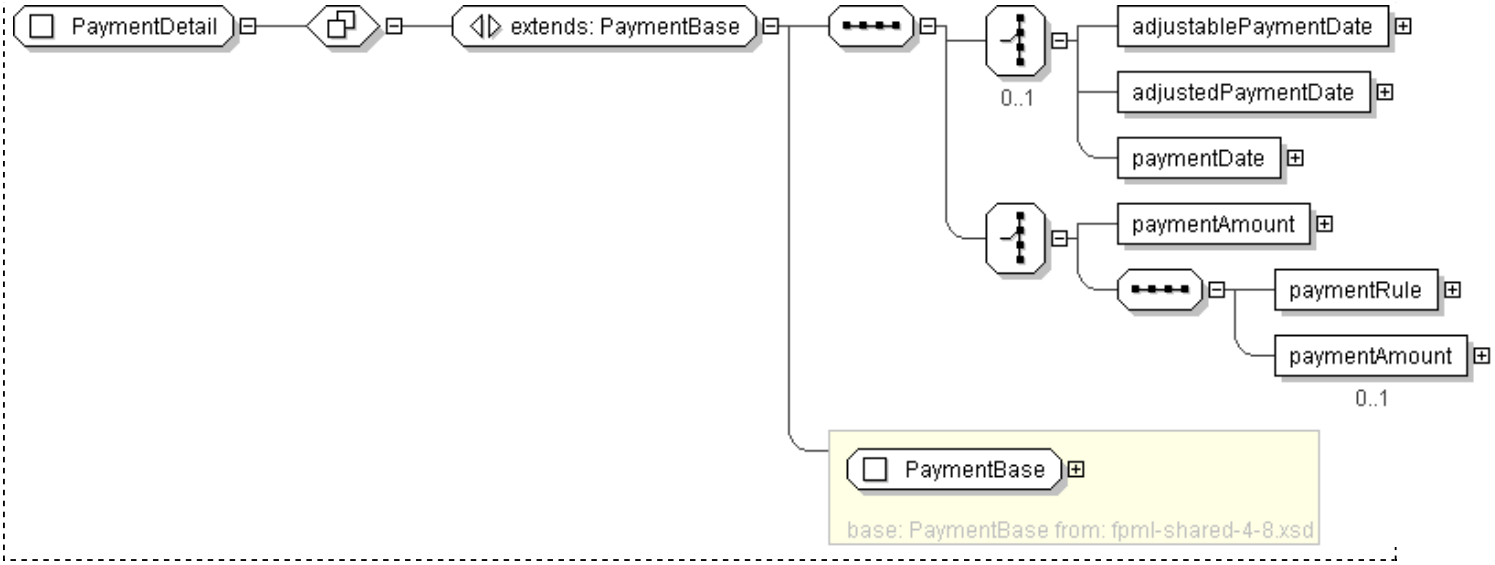
Super-types:	PaymentBase < PaymentDetail (by extension)
Sub-types:	None

Name	PaymentDetail
Used by (from the same schema document)	Complex Type IndependentAmount
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [0..1]
      <adjustablePaymentDate> AdjustableDate2 </adjustablePaymentDate> [1]
      'A fixed amount payment date that shall be subject to adjustment in accordance with
      the applicable business day convention if it would otherwise fall on a day that is
      not a business day. The applicable business day convention and business day are those
      specified in the dateAdjustments element within the generalTerms component. ISDA 2003
      Term: Fixed Rate Payer Payment Date.'

      <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
      'The adjusted payment date. This date should already be adjusted for any applicable
      business day convention. This component is not intended for use in trade confirmation
      but may be specified to allow the fee structure to also serve as a cashflow type
      component.'

      <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
      'Payment date.'

   End Choice
   Start Choice [1]
      <paymentAmount> Money </paymentAmount> [1]
      'A fixed payment amount.'

      <paymentRule> PaymentRule </paymentRule> [1]
      'A type defining the calculation rule.'

      <paymentAmount> Money </paymentAmount> [0..1]
      'A fixed payment amount.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentDetail">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="adjustablePaymentDate" type=" AdjustableDate2 "
          deprecated="true" deprecatedReason="This structure doesn't provide the ability to
          provide a payment date relative to another date. The paymentDate element of type
          AdjustableOrRelativeDate should be used instead. In version 5.0 the date
          structures will be rationalized."/>
          <xsd:element name="adjustedPaymentDate" type=" xsd:date " deprecated="true"
          deprecatedReason="In version 5.0 the date structures will be rationalized."/>
          <xsd:element name="paymentDate" type=" AdjustableOrRelativeDate "/>
        </xsd:choice>
        <xsd:choice>
          <xsd:element name="paymentAmount" type=" Money "/>
          <xsd:sequence>
            <xsd:element name="paymentRule" type=" PaymentRule "/>
            <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
          </xsd:sequence>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentRule

[Table of contents]

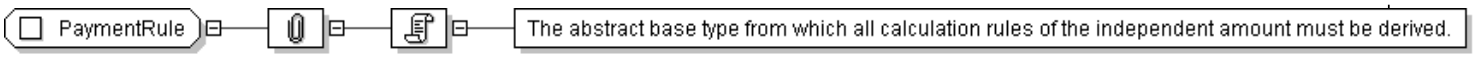
Super-types:	None
Sub-types:	• PercentageRule (by extension)

Name	PaymentRule
Used by (from the same schema document)	Complex Type PaymentDetail
Abstract	yes
Documentation	The abstract base type from which all calculation rules of the independent amount must be derived.

XML Instance Representation

```
<.../>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentRule" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PercentageRule

[Table of contents]

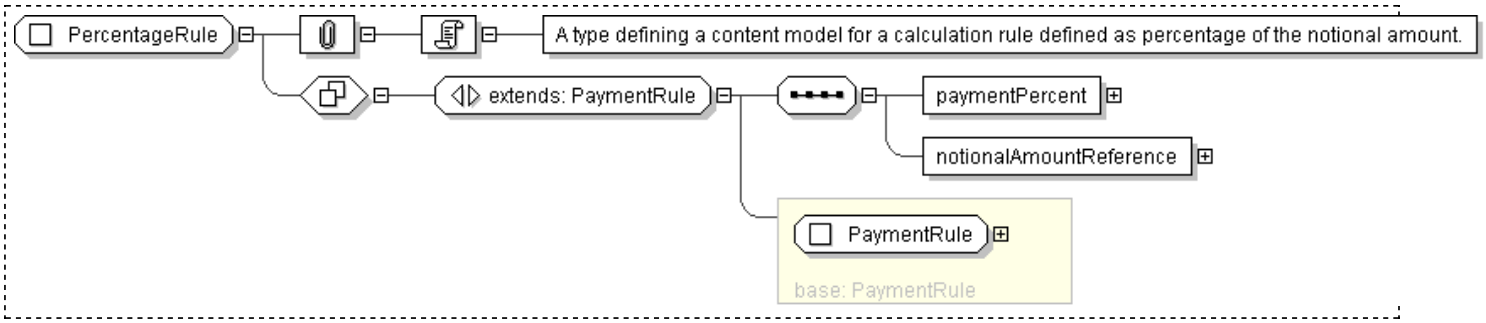
Super-types:	PaymentRule < PercentageRule (by extension)
Sub-types:	None

Name	PercentageRule
Abstract	no
Documentation	A type defining a content model for a calculation rule defined as percentage of the notional amount.

XML Instance Representation

```
<...>
   <paymentPercent> xsd:decimal </paymentPercent> [1]
    'A percentage of the notional amount.'

   <notionalAmountReference> NotionalAmountReference </notionalAmountReference> [1]
    'A reference to the notional amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PercentageRule">
   <xsd:complexContent>
      <xsd:extension base=" PaymentRule ">
         <xsd:sequence>
            <xsd:element name="paymentPercent" type=" xsd:decimal "/>
            <xsd:element name="notionalAmountReference" type=" NotionalAmountReference "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Portfolio

[Table of contents]

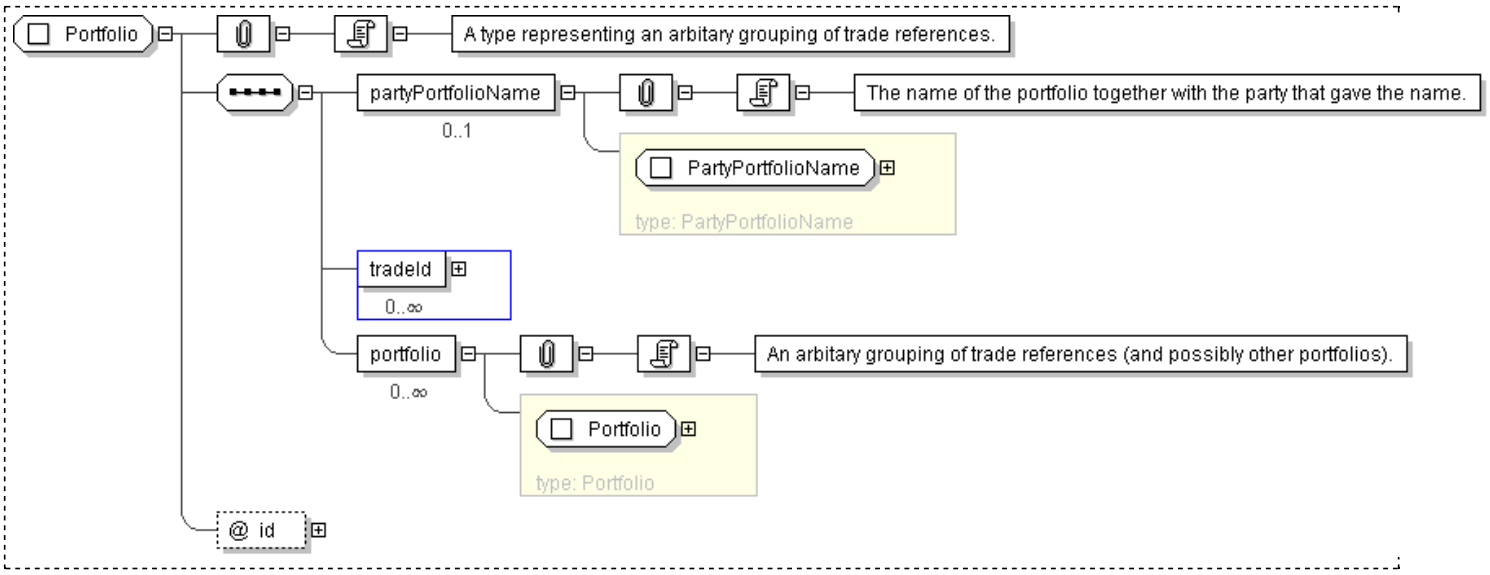
Super-types:	None
Sub-types:	• QueryPortfolio (by extension)

Name	Portfolio
Used by (from the same schema document)	Complex Type DataDocument , Complex Type Portfolio
Abstract	no
Documentation	A type representing an arbitary grouping of trade references.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyPortfolioName> PartyPortfolioName </partyPortfolioName> [0..1]
    'The name of the portfolio together with the party that gave the name.'

   <tradeId> TradeId </tradeId> [0..*]
   <portfolio> Portfolio </portfolio> [0..*]
    'An arbitary grouping of trade references (and possibly other portfolios).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Portfolio">
  <xsd:sequence>
    <xsd:element name="partyPortfolioName" type=" PartyPortfolioName " minOccurs="0"/>
    <xsd:element name="tradeId" type=" TradeId " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="portfolio" type=" Portfolio " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PortfolioName

[Table of contents]

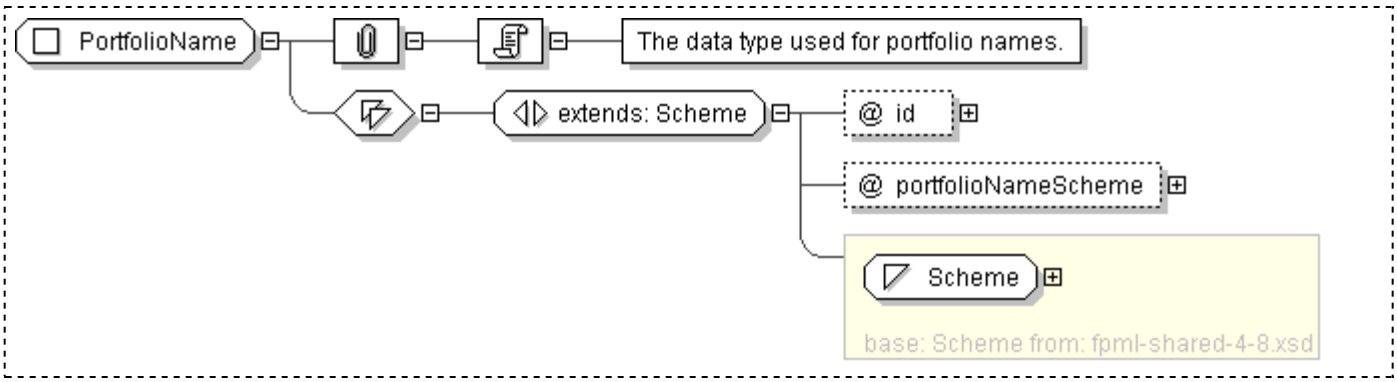
Super-types:	Scheme < PortfolioName (by extension)
Sub-types:	None

Name	PortfolioName
Used by (from the same schema document)	Complex Type PartyPortfolioName
Abstract	no
Documentation	The data type used for portfolio names.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 portfolioNameScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PortfolioName">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="id" type=" xsd:ID "/>
         <xsd:attribute name="portfolioNameScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QueryParameter

[Table of contents]

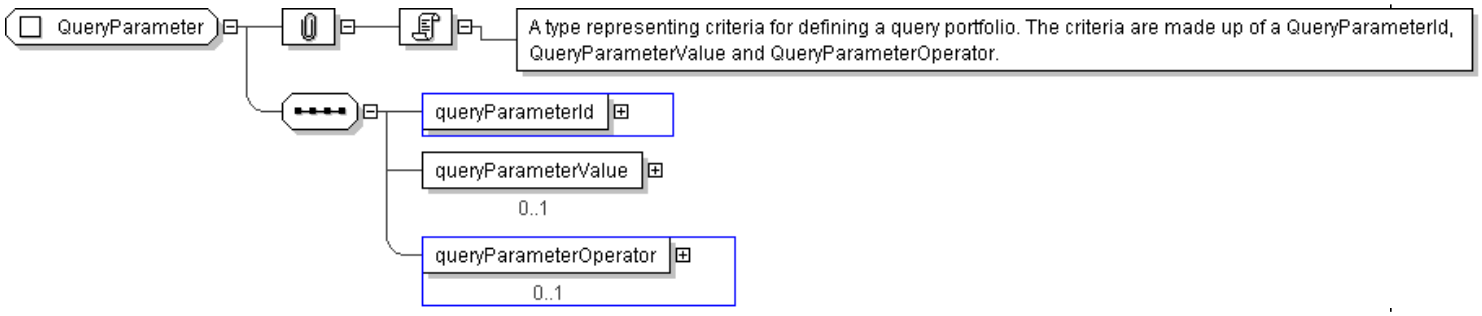
Super-types:	None
Sub-types:	None

Name	QueryParameter
Used by (from the same schema document)	Complex Type QueryPortfolio
Abstract	no
Documentation	A type representing criteria for defining a query portfolio. The criteria are made up of a QueryParameterId, QueryParameterValue and QueryParameterOperator.

XML Instance Representation

```
<...>
   <queryParameterId> QueryParameterId </queryParameterId> [1]
   <queryParameterValue> xsd:normalizedString </queryParameterValue> [0..1]
   <queryParameterOperator> QueryParameterOperator </queryParameterOperator> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QueryParameter">
   <xsd:sequence>
      <xsd:element name="queryParameterId" type=" QueryParameterId "/>
      <xsd:element name="queryParameterValue" type=" xsd:normalizedString " minOccurs="0"/>
      <xsd:element name="queryParameterOperator" type=" QueryParameterOperator " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QueryParameterId

[Table of contents]

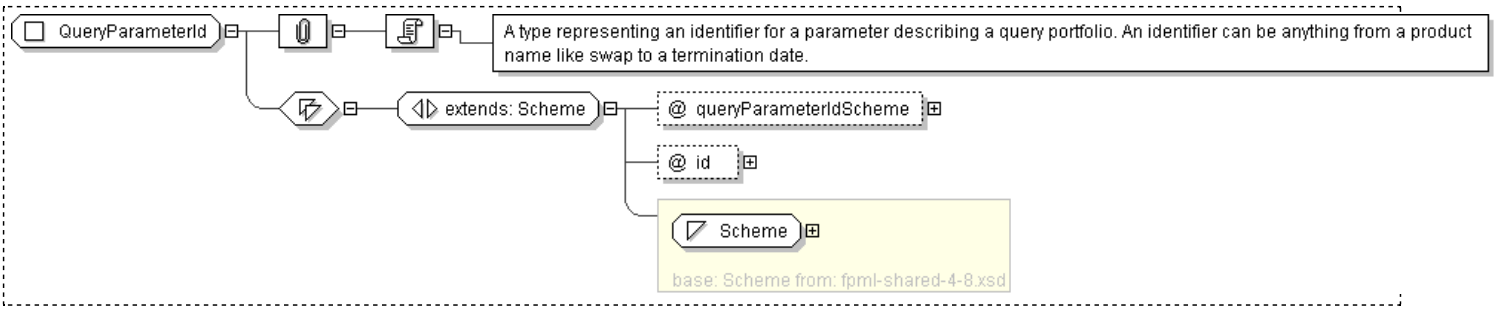
Super-types:	Scheme < QueryParameterId (by extension)
Sub-types:	None

Name	QueryParameterId
Used by (from the same schema document)	Complex Type QueryParameter
Abstract	no
Documentation	A type representing an identifier for a parameter describing a query portfolio. An identifier can be anything from a product name like swap to a termination date.

XML Instance Representation

```
<...
 queryParameterIdScheme=" xsd:anyURI [1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QueryParameterId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="queryParameterIdScheme" type=" xsd:anyURI " use="required"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QueryParameterOperator

[Table of contents]

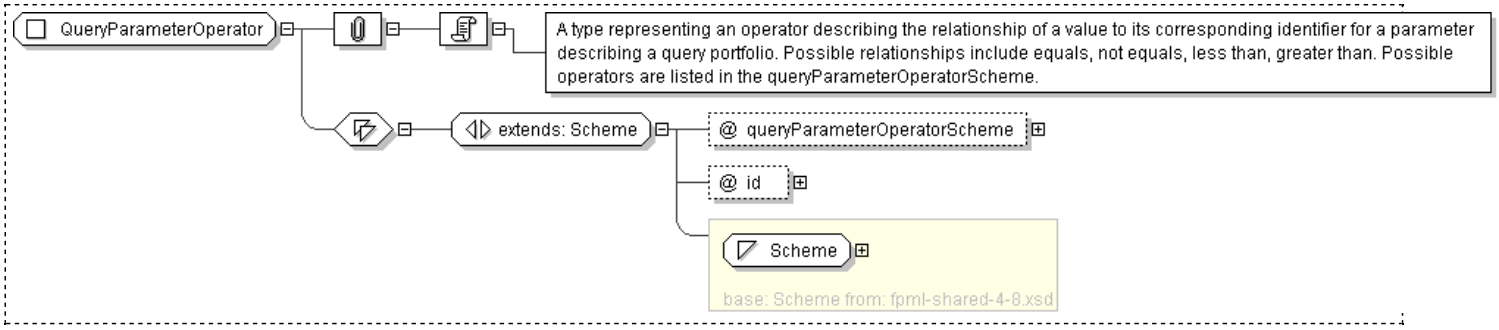
Super-types:	Scheme < QueryParameterOperator (by extension)
Sub-types:	None

Name	QueryParameterOperator
Used by (from the same schema document)	Complex Type QueryParameter
Abstract	no
Documentation	A type representing an operator describing the relationship of a value to its corresponding identifier for a parameter describing a query portfolio. Possible relationships include equals, not equals, less than, greater than. Possible operators are listed in the queryParameterOperatorScheme.

XML Instance Representation

```
<...
 queryParameterOperatorScheme=" xsd:anyURI [0..1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QueryParameterOperator">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="queryParameterOperatorScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/query-parameter-operator"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

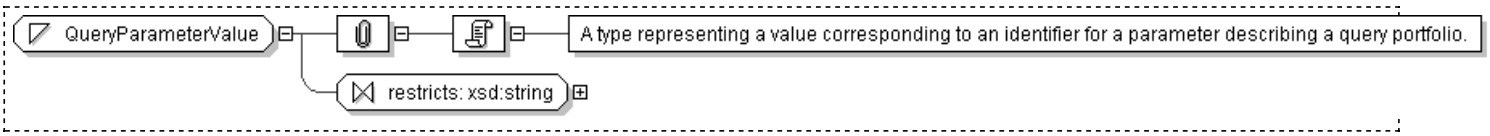
Simple Type: QueryParameterValue

[Table of contents]

Super-types:	xsd:string < QueryParameterValue (by restriction)
Sub-types:	None

Name	QueryParameterValue
Content	• Base XSD Type: string
Documentation	A type representing a value corresponding to an identifier for a parameter describing a query portfolio.

Diagram



Schema Component Representation

```
<xsd:simpleType name="QueryParameterValue">
  <xsd:restriction base=" xsd:string "/>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QueryPortfolio

[Table of contents]

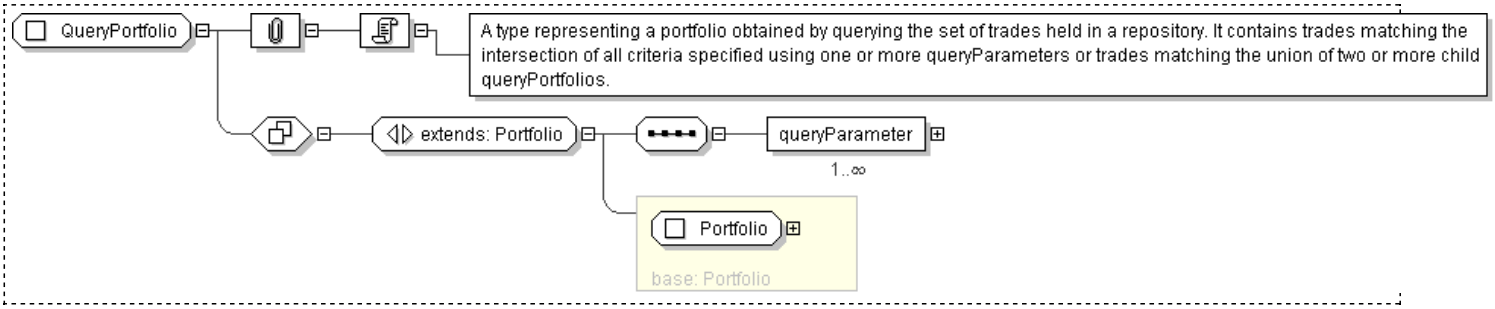
Super-types:	Portfolio < QueryPortfolio (by extension)
Sub-types:	None

Name	QueryPortfolio
Abstract	no
Documentation	A type representing a portfolio obtained by querying the set of trades held in a repository. It contains trades matching the intersection of all criteria specified using one or more queryParameters or trades matching the union of two or more child queryPortfolios.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyPortfolioName> PartyPortfolioName </partyPortfolioName> [0..1]
   'The name of the portfolio together with the party that gave the name.'

   <tradeId> TradeId </tradeId> [0..*]
   <portfolio> Portfolio </portfolio> [0..*]
   'An arbitary grouping of trade references (and possibly other portfolios).'

   <queryParameter> QueryParameter </queryParameter> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QueryPortfolio">
  <xsd:complexContent>
    <xsd:extension base=" Portfolio ">
      <xsd:sequence>
        <xsd:element name="queryParameter" type=" QueryParameter " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Strategy

[Table of contents]

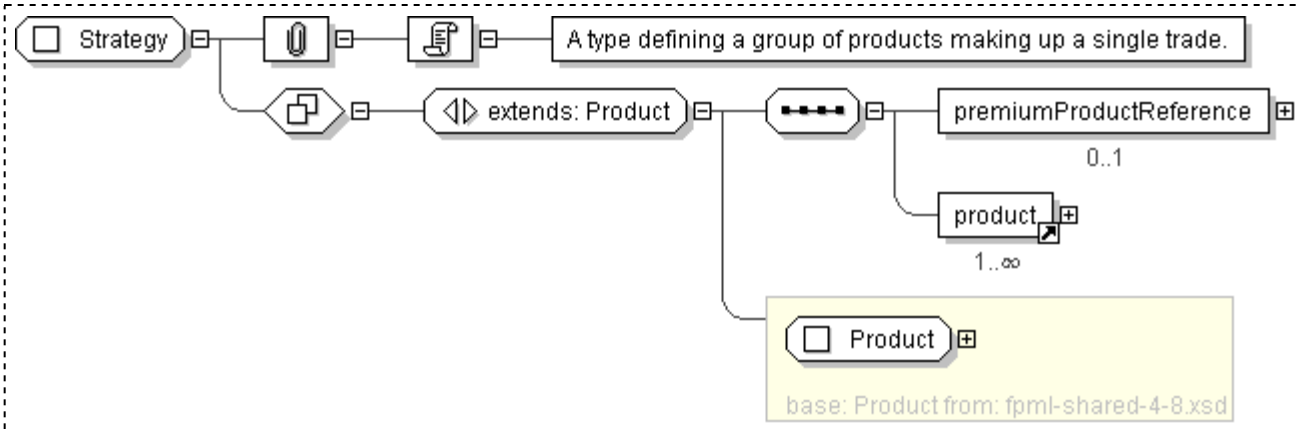
Super-types:	Product < Strategy (by extension)
Sub-types:	None

Name	Strategy
Used by (from the same schema document)	Element strategy
Abstract	no
Documentation	A type defining a group of products making up a single trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <premiumProductReference> ProductReference </premiumProductReference> [0..1]
    'Indicates which product within a strategy represents the premium payment.'

   <product> ... </product> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Strategy">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="premiumProductReference" type=" ProductReference "
            minOccurs="0"/>
            <xsd:element ref=" product " maxOccurs="unbounded"/>
         </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Trade

[Table of contents]

Super-types:	None
Sub-types:	None

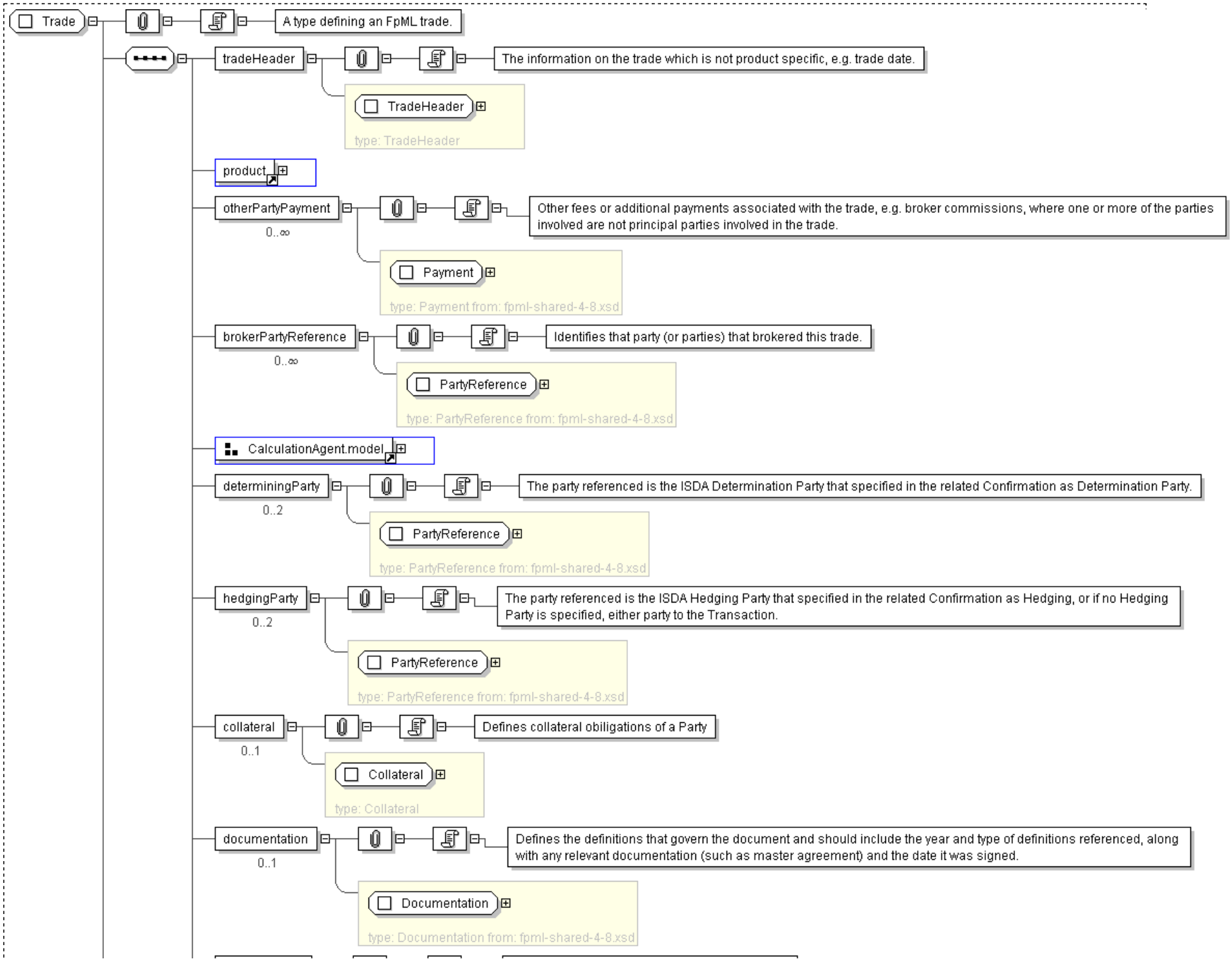
Name	Trade
Used by (from the same schema document)	Complex Type Amendment , Complex Type DataDocument , Model Group TradeOrTradeReference.model
Abstract	no
Documentation	A type defining an FpML trade.

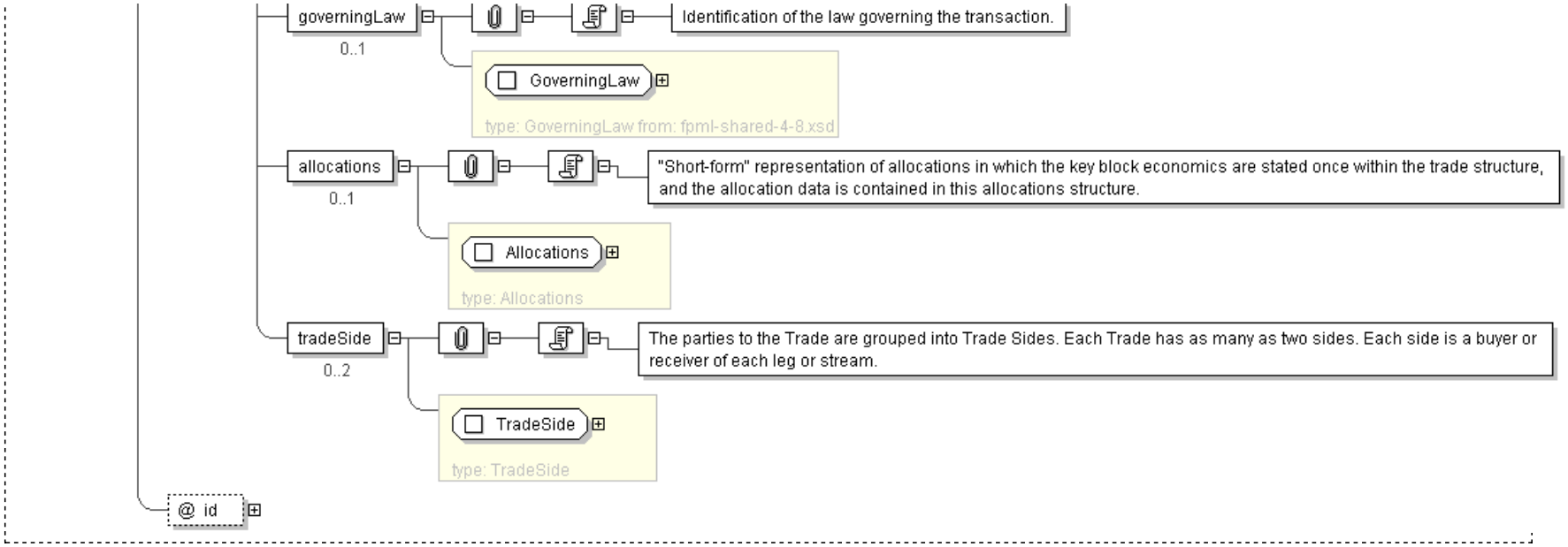
XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <tradeHeader> TradeHeader </tradeHeader> [1]
   'The information on the trade which is not product specific, e.g. trade date.'

   <product> ... </product> [1]
   <otherPartyPayment> Payment </otherPartyPayment> [0..*]
   'Other fees or additional payments associated with the trade, e.g. broker commissions, where one or more of the parties involved
   are not principal parties involved in the trade.'

   <brokerPartyReference> PartyReference </brokerPartyReference> [0..*]
   'Identifies that party (or parties) that brokered this trade.'

   <calculationAgent> CalculationAgent </calculationAgent> [0..1]
   'The ISDA calculation agent responsible for performing duties as defined in the applicable product definitions.'

   <calculationAgentBusinessCenter> BusinessCenter </calculationAgentBusinessCenter> [0..1]
   'The city in which the office through which ISDA Calculation Agent is acting for purposes of the transaction is located The short-
   form confirm for a trade that is executed under a Sovereign or Asia Pacific Master Confirmation Agreement ( MCA ), does not need
   to specify the Calculation Agent. However, the confirm does need to specify the Calculation Agent City. This is due to the fact
   that the MCA sets the value for Calculation Agent but does not set the value for Calculation Agent City.'

   <determiningParty> PartyReference </determiningParty> [0..2]
   'The party referenced is the ISDA Determination Party that specified in the related Confirmation as Determination Party.'

   <hedgingParty> PartyReference </hedgingParty> [0..2]
   'The party referenced is the ISDA Hedging Party that specified in the related Confirmation as Hedging, or if no Hedging Party is
   specified, either party to the Transaction.'

   <collateral> Collateral </collateral> [0..1]
   'Defines collateral obiligations of a Party'

   <documentation> Documentation </documentation> [0..1]
   'Defines the definitions that govern the document and should include the year and type of definitions referenced, along with any
   relevant documentation (such as master agreement) and the date it was signed.'

   <governingLaw> GoverningLaw </governingLaw> [0..1]
   'Identification of the law governing the transaction.'

   <allocations> Allocations </allocations> [0..1]
   '"Short-form" representation of allocations in which the key block economics are stated once within the trade structure, and the
   allocation data is contained in this allocations structure.'

   <tradeSide> TradeSide </tradeSide> [0..2]
   'The parties to the Trade are grouped into Trade Sides. Each Trade has as many as two sides. Each side is a buyer or receiver of
   each leg or stream.'
```

```
</...>
```

Diagram


Trade
A type defining an FpML trade.
tradeHeader
The information on the trade which is not product specific, e.g. trade date.
TradeHeader
type: TradeHeader
product
otherPartyPayment
0..∞
Other fees or additional payments associated with the trade, e.g. broker commissions, where one or more of the parties involved are not principal parties involved in the trade.
Payment
type: Payment from: fpml-shared-4-8.xsd
brokerPartyReference
0..∞
Identifies that party (or parties) that brokered this trade.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
CalculationAgent.model
determiningParty
0..2
The party referenced is the ISDA Determination Party that specified in the related Confirmation as Determination Party.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
hedgingParty
0..2
The party referenced is the ISDA Hedging Party that specified in the related Confirmation as Hedging, or if no Hedging Party is specified, either party to the Transaction.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
collateral
0..1
Defines collateral obiligations of a Party
Collateral
type: Collateral
documentation
0..1
Defines the definitions that govern the document and should include the year and type of definitions referenced, along with any relevant documentation (such as master agreement) and the date it was signed.
Documentation
type: Documentation from: fpml-shared-4-8.xsd



Schema Component Representation

```
<xsd:complexType name="Trade">
  <xsd:sequence>
    <xsd:element name="tradeHeader" type=" TradeHeader "/>
    <xsd:element ref=" product "/>
    <xsd:element name="otherPartyPayment" type=" Payment " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="brokerPartyReference" type=" PartyReference " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:group ref=" CalculationAgent.model "/>
    <xsd:element name="determiningParty" type=" PartyReference " minOccurs="0" maxOccurs="2"/>
    <xsd:element name="hedgingParty" type=" PartyReference " minOccurs="0" maxOccurs="2"/>
    <xsd:element name="collateral" type=" Collateral " minOccurs="0"/>
    <xsd:element name="documentation" type=" Documentation " minOccurs="0"/>
    <xsd:element name="governingLaw" type=" GoverningLaw " minOccurs="0"/>
    <xsd:element name="allocations" type=" Allocations " minOccurs="0"/>
    <xsd:element name="tradeSide" type=" TradeSide " minOccurs="0" maxOccurs="2"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeDifference

[Table of contents]

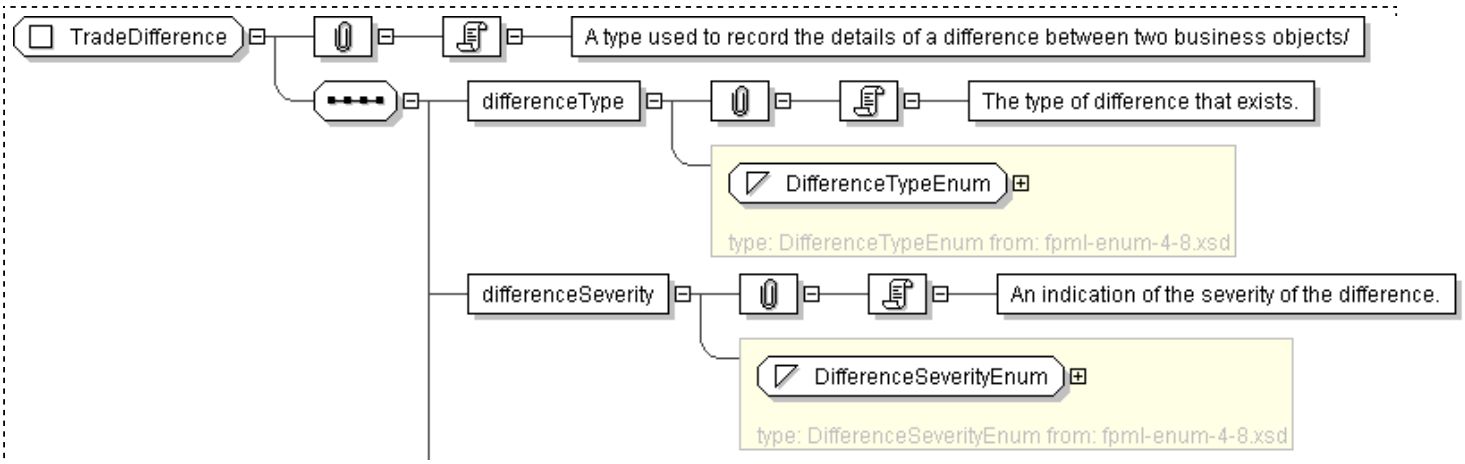
Super-types:	None
Sub-types:	None

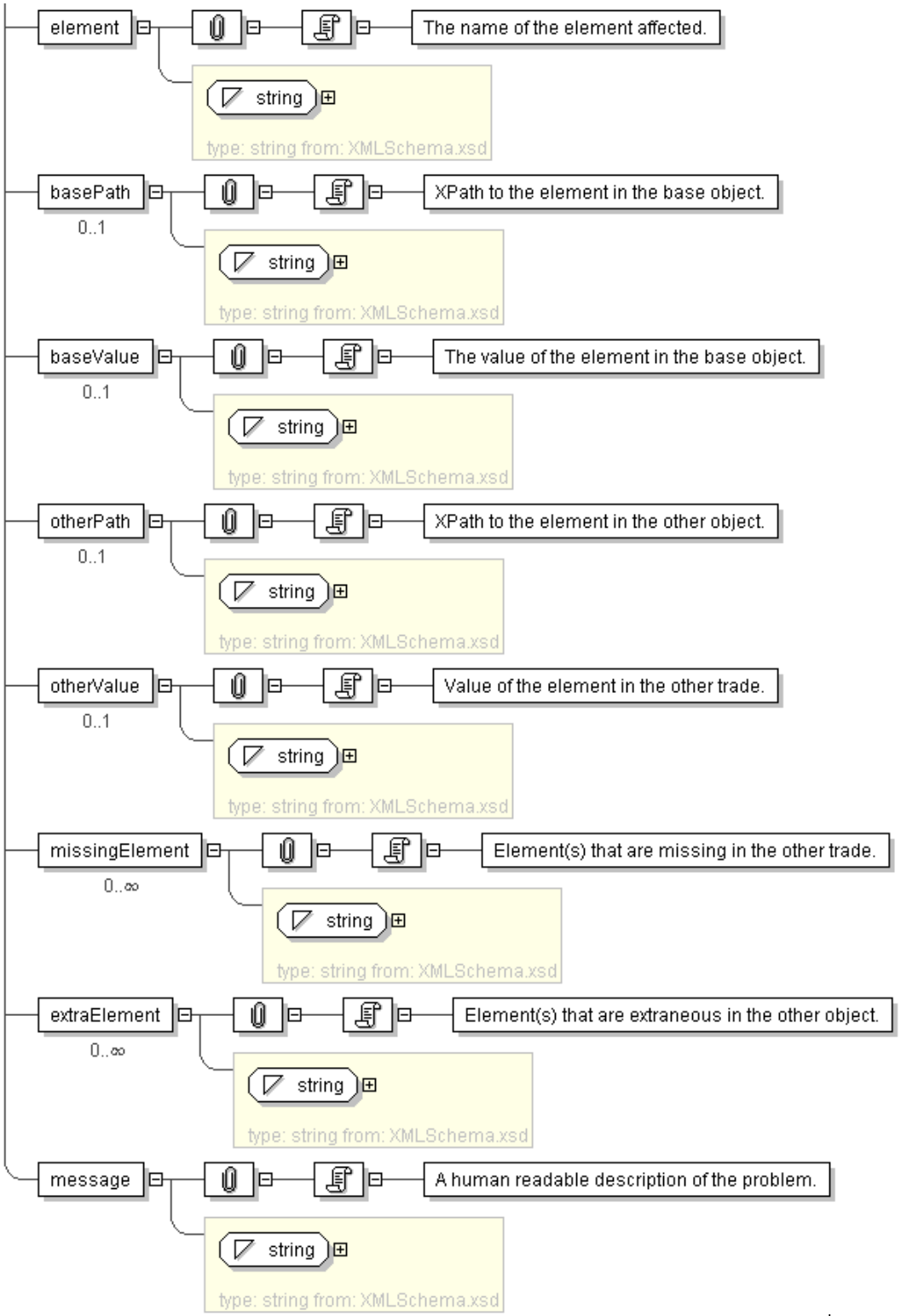
Name	TradeDifference
Used by (from the same schema document)	Complex Type BestFitTrade
Abstract	no
Documentation	A type used to record the details of a difference between two business objects/

XML Instance Representation

```
<...>
  <differenceType> DifferenceTypeEnum </differenceType> [1]
  'The type of difference that exists.'

  <differenceSeverity> DifferenceSeverityEnum </differenceSeverity> [1]
  'An indication of the severity of the difference.'

  <element> xsd:string </element> [1]
  'The name of the element affected.'

  <basePath> xsd:string </basePath> [0..1]
  'XPath to the element in the base object.'

  <baseValue> xsd:string </baseValue> [0..1]
  'The value of the element in the base object.'

  <otherPath> xsd:string </otherPath> [0..1]
  'XPath to the element in the other object.'

  <otherValue> xsd:string </otherValue> [0..1]
  'Value of the element in the other trade.'

  <missingElement> xsd:string </missingElement> [0..*]
  'Element(s) that are missing in the other trade.'

  <extraElement> xsd:string </extraElement> [0..*]
  'Element(s) that are extraneous in the other object.'

  <message> xsd:string </message> [1]
  'A human readable description of the problem.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="TradeDifference">
  <xsd:sequence>
    <xsd:element name="differenceType" type=" DifferenceTypeEnum "/>
    <xsd:element name="differenceSeverity" type=" DifferenceSeverityEnum "/>
    <xsd:element name="element" type=" xsd:string "/>
    <xsd:element name="basePath" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="baseValue" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="otherPath" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="otherValue" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="missingElement" type=" xsd:string " minOccurs="0"
    maxOccurs="unbounded"/>
    <xsd:element name="extraElement" type=" xsd:string " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="message" type=" xsd:string "/>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeHeader

[Table of contents]

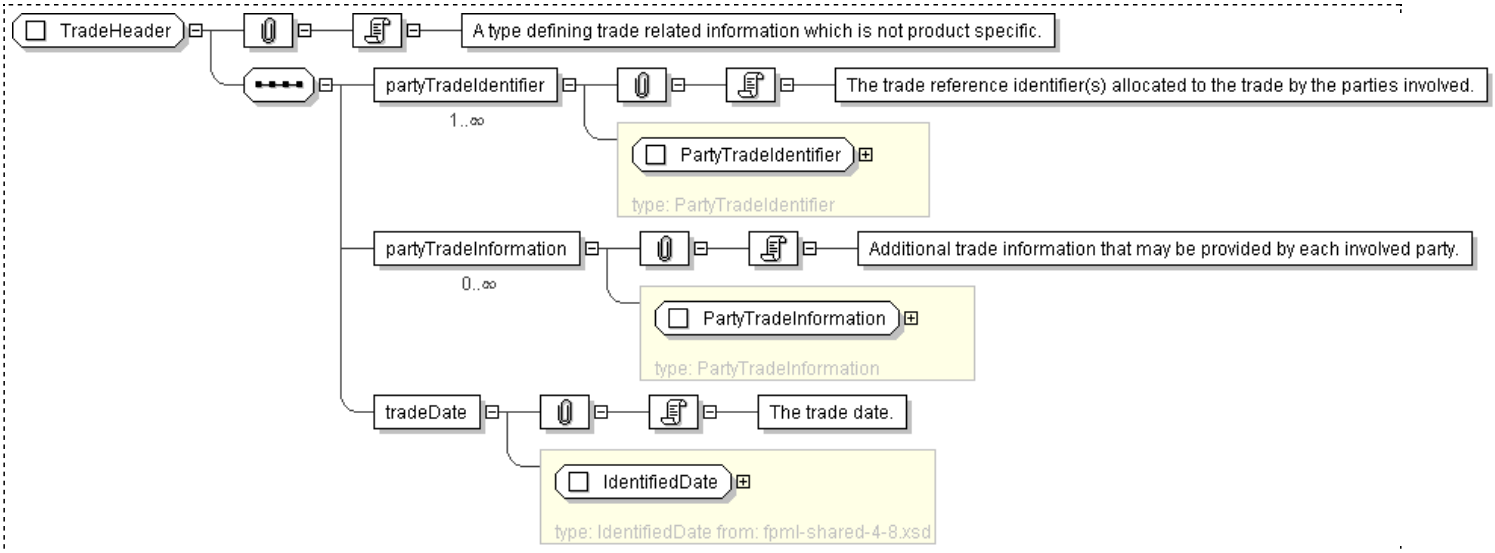
Super-types:	None
Sub-types:	None

Name	TradeHeader
Used by (from the same schema document)	Complex Type Trade
Abstract	no
Documentation	A type defining trade related information which is not product specific.

XML Instance Representation

```
<...>
  <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1..*]
  'The trade reference identifier(s) allocated to the trade by the parties involved.'

  <partyTradeInformation> PartyTradeInformation </partyTradeInformation> [0..*]
  'Additional trade information that may be provided by each involved party.'

  <tradeDate> IdentifiedDate </tradeDate> [1]
  'The trade date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeHeader">
  <xsd:sequence>
    <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier " maxOccurs="unbounded"/>
    <xsd:element name="partyTradeInformation" type=" PartyTradeInformation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="tradeDate" type=" IdentifiedDate "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeId

[Table of contents]

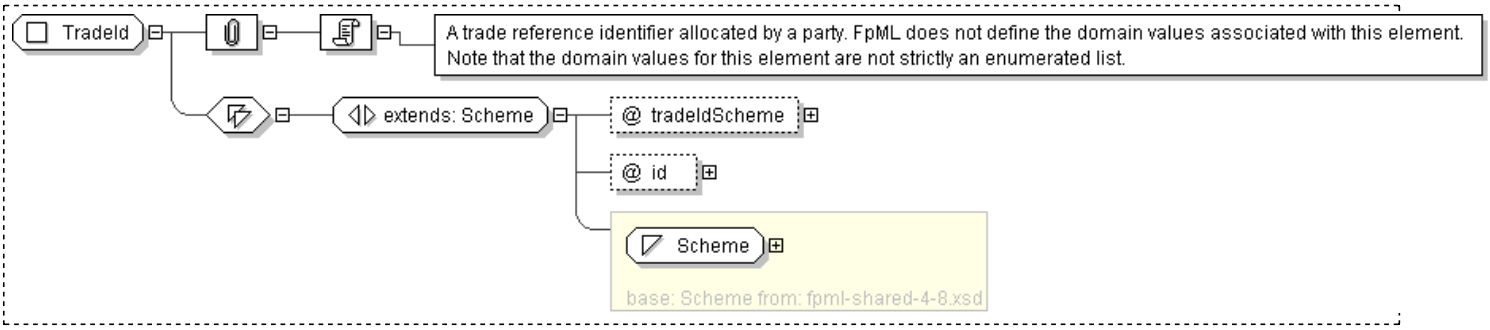
Super-types:	Scheme < TradeId (by extension)
Sub-types:	None

Name	TradeId
Used by (from the same schema document)	Complex Type Portfolio , Complex Type TradeIdentifier , Complex Type VersionedTradeId
Abstract	no
Documentation	A trade reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not strictly an enumerated list.

XML Instance Representation

```
<...
 tradeIdScheme=" xsd:anyURI [1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="tradeIdScheme" type=" xsd:anyURI " use="required"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeIdentifier

[Table of contents]

Super-types: None

Sub-types:

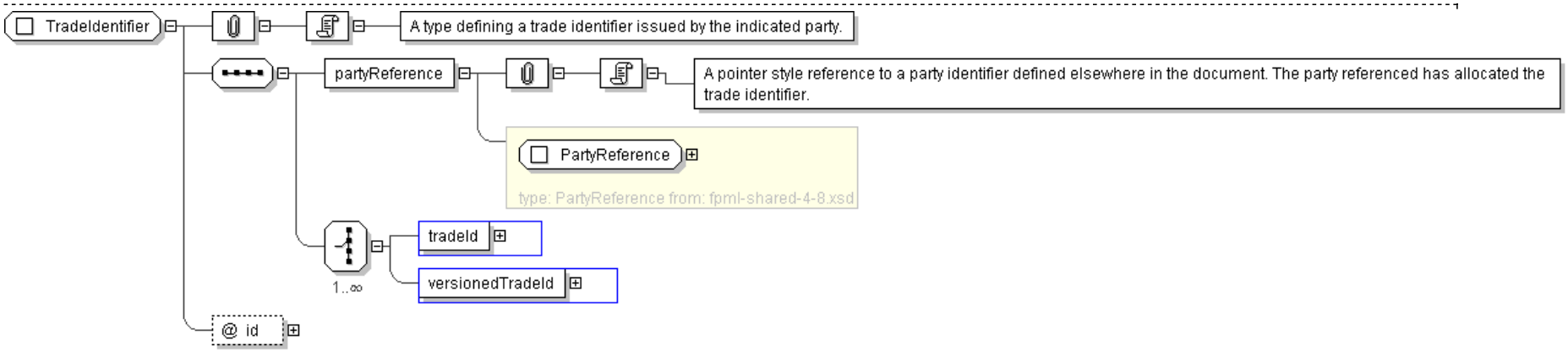
- PartyTradeIdentifier (by extension)
  - AllocationTradeIdentifier (by extension)
  - BlockTradeIdentifier (by extension)

Name	TradeIdentifier
Used by (from the same schema document)	Complex Type BestFitTrade
Abstract	no
Documentation	A type defining a trade identifier issued by the indicated party.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <partyReference> PartyReference </partyReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced has allocated the trade
   identifier.'

   Start Choice [1..*]
      <tradeId> TradeId </tradeId> [1]
      <versionedTradeId> VersionedTradeId </versionedTradeId> [1]
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeIdentifier">
   <xsd:sequence>
      <xsd:element name="partyReference" type=" PartyReference "/>
      <xsd:choice maxOccurs="unbounded">
         <xsd:element name="tradeId" type=" TradeId "/>
         <xsd:element name="versionedTradeId" type=" VersionedTradeId "/>
      </xsd:choice>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Trader

[Table of contents]

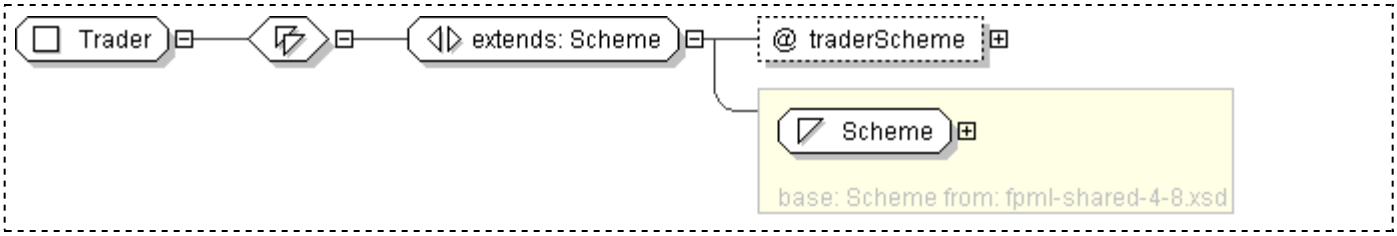
Super-types:	Scheme < Trader (by extension)
Sub-types:	None

Name	Trader
Used by (from the same schema document)	Complex Type PartyTradeInformation
Abstract	no

XML Instance Representation

```
<...
 traderScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Trader">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="traderScheme" type=" xsd:anyURI " use="optional"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: TradeSide

[Table of contents]

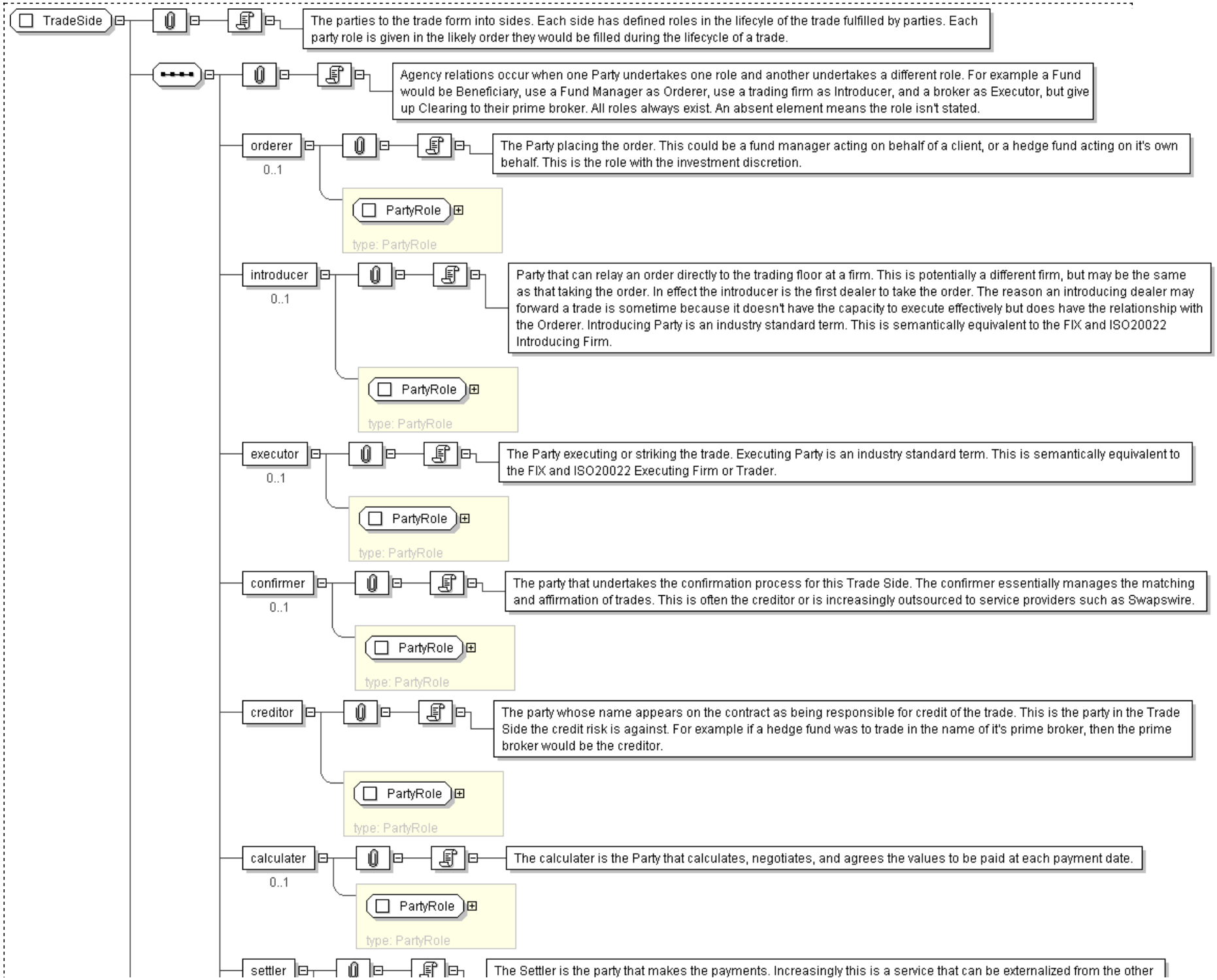
Super-types:	None
Sub-types:	None

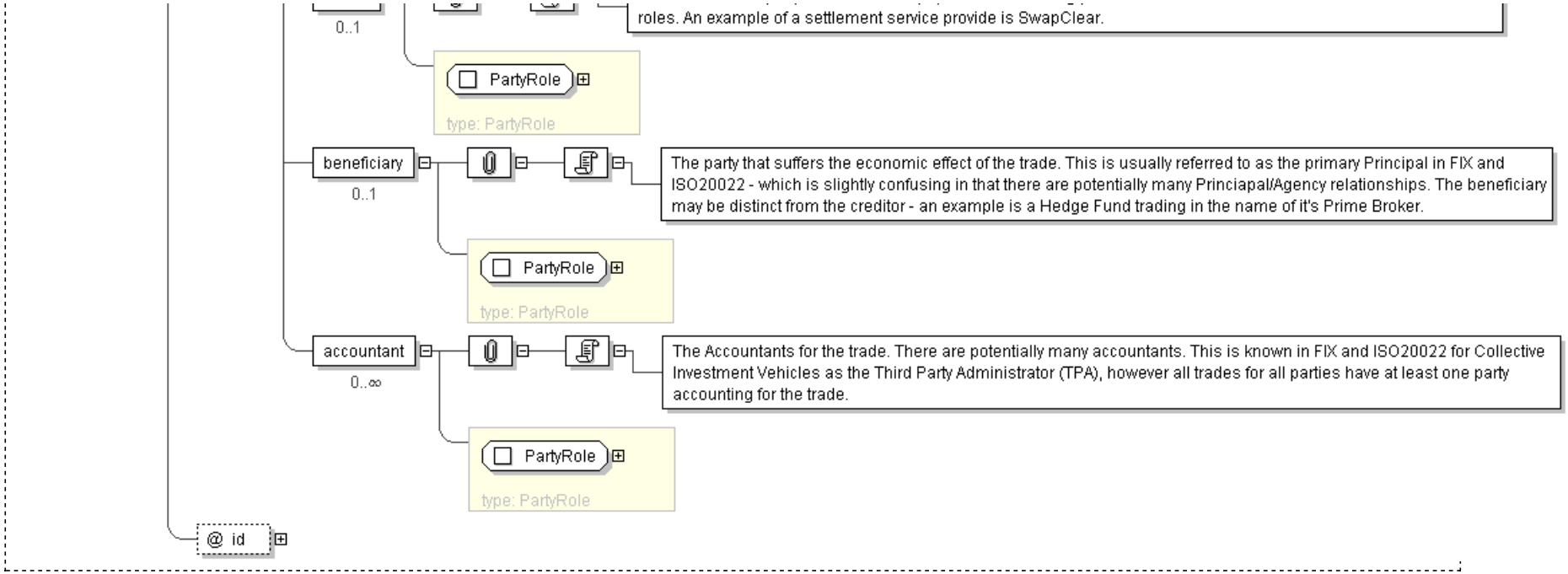
Name	TradeSide
Used by (from the same schema document)	Complex Type Trade
Abstract	no
Documentation	The parties to the trade form into sides. Each side has defined roles in the lifecyle of the trade fulfilled by parties. Each party role is given in the likely order they would be filled during the lifecycle of a trade.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
   <orderer> PartyRole </orderer> [0..1]
   'The Party placing the order. This could be a fund manager acting on behalf of a client, or a hedge fund acting on it\'s own
   behalf. This is the role with the investment discretion.'

   <introducer> PartyRole </introducer> [0..1]
   'Party that can relay an order directly to the trading floor at a firm. This is potentially a different firm, but may be the
   same as that taking the order. In effect the introducer is the first dealer to take the order. The reason an introducing dealer
   may forward a trade is sometime because it doesn\'t have the capacity to execute effectively but does have the relationship with
   the Orderer. Introducing Party is an industry standard term. This is semantically equivalent to the FIX and ISO20022 Introducing
   Firm.'

   <executor> PartyRole </executor> [0..1]
   'The Party executing or striking the trade. Executing Party is an industry standard term. This is semantically equivalent to the
   FIX and ISO20022 Executing Firm or Trader.'

   <confirmer> PartyRole </confirmer> [0..1]
   'The party that undertakes the confirmation process for this Trade Side. The confirmer essentially manages the matching and
   affirmation of trades. This is often the creditor or is increasingly outsourced to service providers such as Swapswire.'

   <creditor> PartyRole </creditor> [1]
   'The party whose name appears on the contract as being responsible for credit of the trade. This is the party in the Trade Side
   the credit risk is against. For example if a hedge fund was to trade in the name of it\'s prime broker, then the prime broker
   would be the creditor.'

   <calculater> PartyRole </calculater> [0..1]
   'The calculater is the Party that calculates, negotiates, and agrees the values to be paid at each payment date.'

   <settler> PartyRole </settler> [0..1]
   'The Settler is the party that makes the payments. Increasingly this is a service that can be externalized from the other roles.
   An example of a settlement service provide is SwapClear.'

   <beneficiary> PartyRole </beneficiary> [0..1]
   'The party that suffers the economic effect of the trade. This is usually referred to as the primary Principal in FIX and
   ISO20022 - which is slightly confusing in that there are potentially many Princiapal/Agency relationships. The beneficiary may be
   distinct from the creditor - an example is a Hedge Fund trading in the name of it\'s Prime Broker.'

   <accountant> PartyRole </accountant> [0..*]
   'The Accountants for the trade. There are potentially many accountants. This is known in FIX and ISO20022 for Collective
   Investment Vehicles as the Third Party Administrator (TPA), however all trades for all parties have at least one party
   accounting for the trade.'

</...>
```

Diagram


TradeSide
The parties to the trade form into sides. Each side has defined roles in the lifecyle of the trade fulfilled by parties. Each party role is given in the likely order they would be filled during the lifecycle of a trade.
Agency relations occur when one Party undertakes one role and another undertakes a different role. For example a Fund would be Beneficiary, use a Fund Manager as Orderer, use a trading firm as Introducer, and a broker as Executor, but give up Clearing to their prime broker. All roles always exist. An absent element means the role isn't stated.
orderer
0..1
The Party placing the order. This could be a fund manager acting on behalf of a client, or a hedge fund acting on it's own behalf. This is the role with the investment discretion.
PartyRole
type: PartyRole
introducer
0..1
Party that can relay an order directly to the trading floor at a firm. This is potentially a different firm, but may be the same as that taking the order. In effect the introducer is the first dealer to take the order. The reason an introducing dealer may forward a trade is sometime because it doesn't have the capacity to execute effectively but does have the relationship with the Orderer. Introducing Party is an industry standard term. This is semantically equivalent to the FIX and ISO20022 Introducing Firm.
PartyRole
type: PartyRole
executor
0..1
The Party executing or striking the trade. Executing Party is an industry standard term. This is semantically equivalent to the FIX and ISO20022 Executing Firm or Trader.
PartyRole
type: PartyRole
confirmer
0..1
The party that undertakes the confirmation process for this Trade Side. The confirmer essentially manages the matching and affirmation of trades. This is often the creditor or is increasingly outsourced to service providers such as Swapswire.
PartyRole
type: PartyRole
creditor
The party whose name appears on the contract as being responsible for credit of the trade. This is the party in the Trade Side the credit risk is against. For example if a hedge fund was to trade in the name of it's prime broker, then the prime broker would be the creditor.
PartyRole
type: PartyRole
calculater
0..1
The calculater is the Party that calculates, negotiates, and agrees the values to be paid at each payment date.
PartyRole
type: PartyRole
settler
The Settler is the party that makes the payments. Increasingly this is a service that can be externalized from the other



Schema Component Representation

```
<xsd:complexType name="TradeSide">
  <xsd:sequence>
    <xsd:element name="orderer" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="introducer" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="executor" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="confirmer" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="creditor" type=" PartyRole "/>
    <xsd:element name="calculater" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="settler" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="beneficiary" type=" PartyRole " minOccurs="0"/>
    <xsd:element name="accountant" type=" PartyRole " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Validation

[Table of contents]

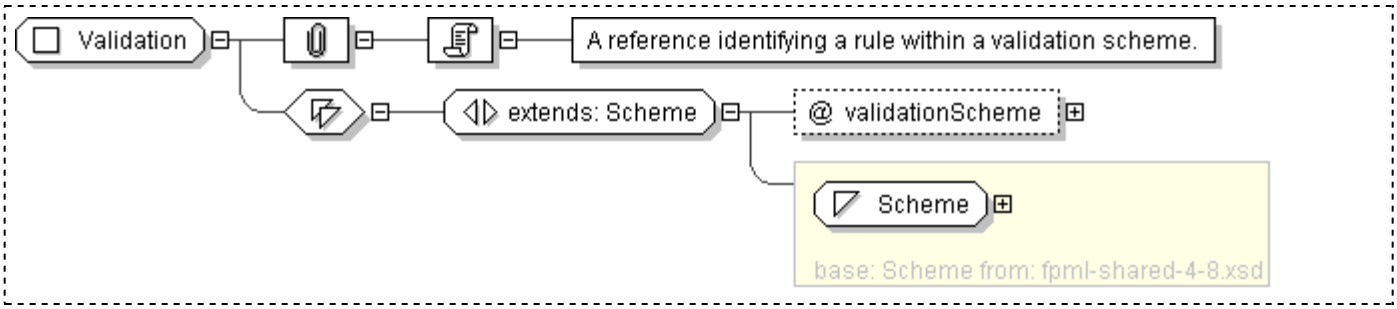
Super-types:	Scheme < Validation (by extension)
Sub-types:	None

Name	Validation
Used by (from the same schema document)	Model Group Validation.model
Abstract	no
Documentation	A reference identifying a rule within a validation scheme.

XML Instance Representation

```
<...
 validationScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Validation">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="validationScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VersionedContractId

[Table of contents]

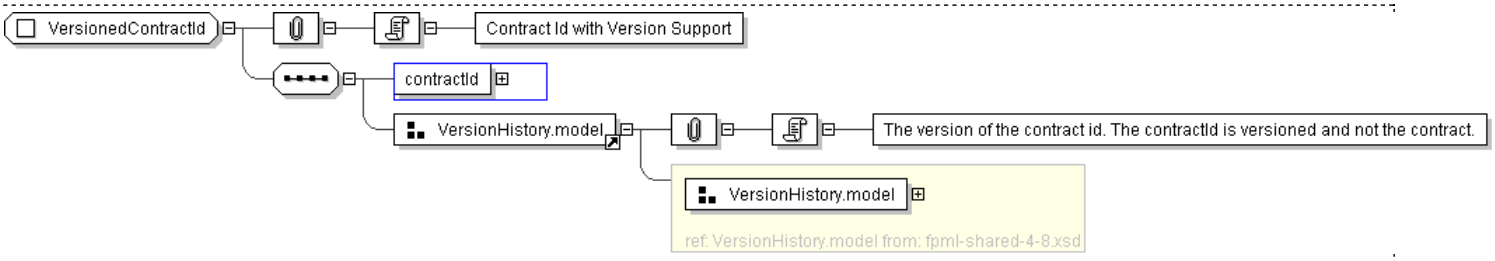
Super-types:	None
Sub-types:	None

Name	VersionedContractId
Used by (from the same schema document)	Complex Type ContractIdentifier
Abstract	no
Documentation	Contract Id with Version Support

XML Instance Representation

```
<...>
   <contractId> ContractId </contractId> [1]
   <version> xsd:nonNegativeInteger </version> [1]
   'The version number'

   <effectiveDate> IdentifiedDate </effectiveDate> [0..1]
   'Optionally it is possible to specify a version effective date when a versionId is supplied.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VersionedContractId">
   <xsd:sequence>
      <xsd:element name="contractId" type=" ContractId "/>
      <xsd:group ref=" VersionHistory.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VersionedTradeId

[Table of contents]

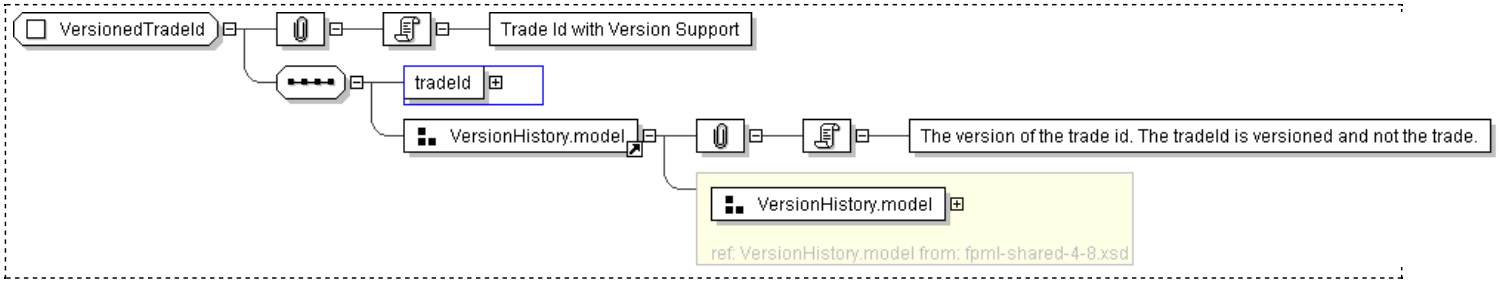
Super-types:	None
Sub-types:	None

Name	VersionedTradeId
Used by (from the same schema document)	Complex Type TradeIdentifier
Abstract	no
Documentation	Trade Id with Version Support

XML Instance Representation

```
<...>
   <tradeId> TradeId </tradeId> [1]
   <version> xsd:nonNegativeInteger </version> [1]
   'The version number'

   <effectiveDate> IdentifiedDate </effectiveDate> [0..1]
   'Optionally it is possible to specify a version effective date when a versionId is supplied.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VersionedTradeId">
   <xsd:sequence>
      <xsd:element name="tradeId" type=" TradeId "/>
      <xsd:group ref=" VersionHistory.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Simple Type: AveragingInOutEnum
  - Simple Type: AveragingMethodEnum
  - Simple Type: BreakageCostEnum
  - Simple Type: BullionTypeEnum
  - Simple Type: BusinessDayConventionEnum
  - Simple Type: CalculationAgentPartyEnum
  - Simple Type: CommissionDenominationEnum
  - Simple Type: CommodityBullionSettlementDisruptionEnum
  - Simple Type: CommodityDayTypeEnum
  - Simple Type: CompoundingMethodEnum
  - Simple Type: ConditionsPrecedentEnum
  - Simple Type: DayOfWeekEnum
  - Simple Type: DayTypeEnum
  - Simple Type: DeliveryDatesEnum
  - Simple Type: DeliveryTypeEnum
  - Simple Type: DifferenceSeverityEnum
  - Simple Type: DifferenceTypeEnum
  - Simple Type: DiscountingTypeEnum
  - Simple Type: DisruptionFallbacksEnum
  - Simple Type: DividendAmountTypeEnum
  - Simple Type: DividendCompositionEnum
  - Simple Type: DividendDateReferenceEnum
  - Simple Type: DividendEntitlementEnum
  - Simple Type: DividendPeriodEnum
  - Simple Type: DrawdownEventTypeEnum
  - Simple Type: ElectricityProductTypeEnum
  - Simple Type: ExerciseStyleEnum
  - Simple Type: FPVFinalPriceElectionFallbackEnum
  - Simple Type: FeeElectionEnum
  - Simple Type: FlatRateEnum
  - Simple Type: FraDiscountingEnum
  - Simple Type: FrequencyTypeEnum
  - Simple Type: FxBarrierTypeEnum
  - Simple Type: GasProductTypeEnum
  - Simple Type: IndexEventConsequenceEnum
  - Simple Type: InterestCalculationMethodEnum
  - Simple Type: InterestPaidWithRepaymentEnum
  - Simple Type: InterestShortfallCapEnum
  - Simple Type: InterpolationPeriodEnum
  - Simple Type: LcPurposeEnum
  - Simple Type: LcTypeEnum
  - Simple Type: LengthUnitEnum
  - Simple Type: LoanRepaymentConfirmEnum
  - Simple Type: MarketDisruptionEventsEnum
  - Simple Type: MethodOfAdjustmentEnum
  - Simple Type: NationalisationOrInsolvencyOrDelistingEventEnum
  - Simple Type: NegativeInterestRateTreatmentEnum
  - Simple Type: NonCashDividendTreatmentEnum
  - Simple Type: NotionalAdjustmentEnum
  - Simple Type: ObligationCategoryEnum
  - Simple Type: OnGoingFeeTypeEnum
  - Simple Type: OneOffFeeTypeEnum

- Simple Type: OptionTypeEnum
- Simple Type: PayRelativeToEnum
- Simple Type: PayerReceiverEnum
- Simple Type: PayoutEnum
- Simple Type: PeriodEnum
- Simple Type: PeriodExtendedEnum
- Simple Type: PremiumQuoteBasisEnum
- Simple Type: PremiumTypeEnum
- Simple Type: PriceExpressionEnum
- Simple Type: PutCallEnum
- Simple Type: QuotationRateTypeEnum
- Simple Type: QuotationSideEnum
- Simple Type: QuotationStyleEnum
- Simple Type: QuoteBasisEnum
- Simple Type: RateTreatmentEnum
- Simple Type: RealisedVarianceMethodEnum
- Simple Type: ResetRelativeToEnum
- Simple Type: ReturnTypeEnum
- Simple Type: RollConventionEnum
- Simple Type: RoundingDirectionEnum
- Simple Type: SettlementPeriodDurationEnum
- Simple Type: SettlementTypeEnum
- Simple Type: ShareExtraordinaryEventEnum
- Simple Type: SideRateBasisEnum
- Simple Type: SpecifiedPriceEnum
- Simple Type: StandardSettlementStyleEnum
- Simple Type: StepRelativeToEnum
- Simple Type: StrikeQuoteBasisEnum
- Simple Type: StubPeriodTypeEnum
- Simple Type: TimeTypeEnum
- Simple Type: TouchConditionEnum
- Simple Type: TriggerConditionEnum
- Simple Type: TriggerTimeTypeEnum
- Simple Type: TriggerTypeEnum
- Simple Type: ValuationMethodEnum
- Simple Type: WeeklyRollConventionEnum

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 7129 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore

xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 7129 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

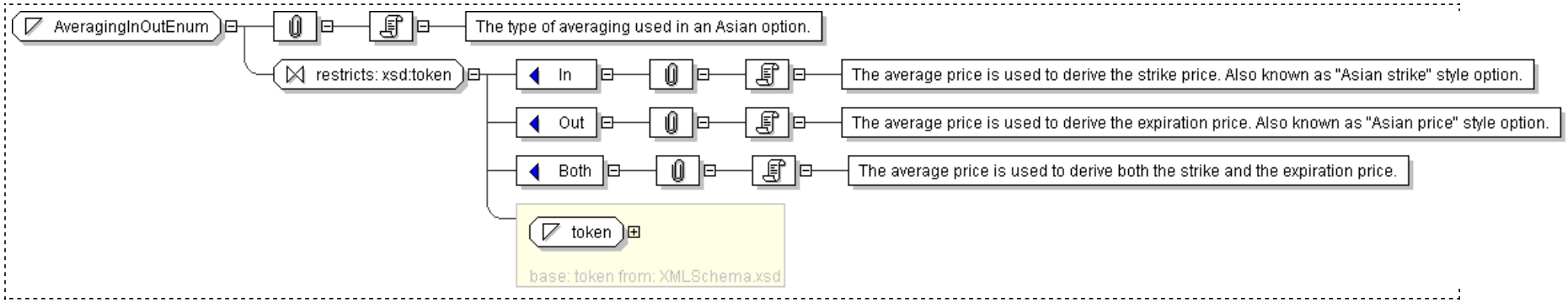
Simple Type: AveragingInOutEnum

[Table of contents]

Super-types:	xsd:token < AveragingInOutEnum (by restriction)
Sub-types:	None

Name	AveragingInOutEnum
Content	• Base XSD Type: token • *value* comes from list: {'In'|'Out'|'Both'}
Documentation	The type of averaging used in an Asian option.

Diagram



Schema Component Representation

```
<xsd:simpleType name="AveragingInOutEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="In"/>
    <xsd:enumeration value="Out"/>
    <xsd:enumeration value="Both"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

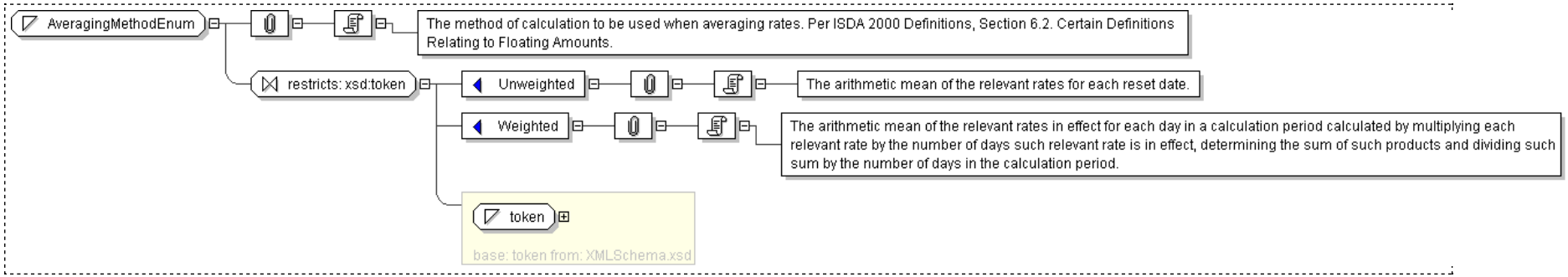
Simple Type: AveragingMethodEnum

[Table of contents]

Super-types:	xsd:token < AveragingMethodEnum (by restriction)
Sub-types:	None

Name	AveragingMethodEnum
Content	• Base XSD Type: token • *value* comes from list: {'Unweighted'|'Weighted'}
Documentation	The method of calculation to be used when averaging rates. Per ISDA 2000 Definitions, Section 6.2. Certain Definitions Relating to Floating Amounts.

Diagram



Schema Component Representation

```
<xsd:simpleType name="AveragingMethodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Unweighted"/>
    <xsd:enumeration value="Weighted"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

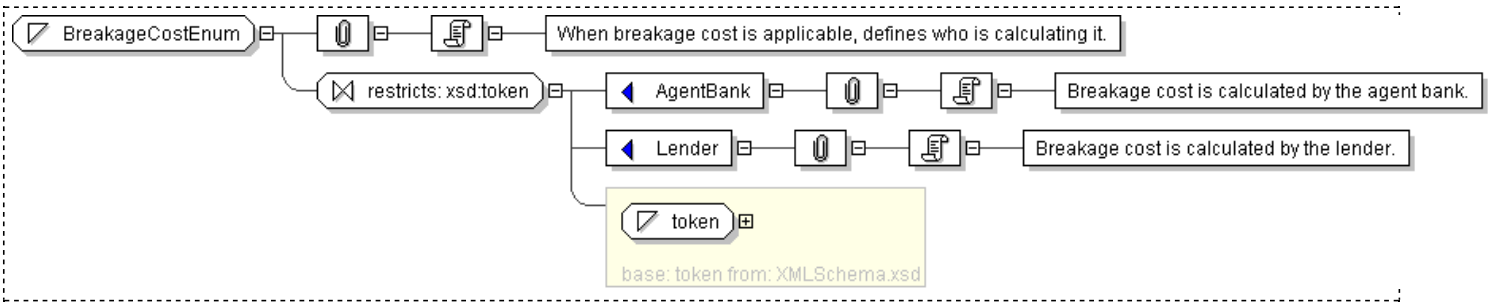
Simple Type: BreakageCostEnum

[Table of contents]

Super-types:	xsd:token < BreakageCostEnum (by restriction)
Sub-types:	None

Name	BreakageCostEnum
Content	• Base XSD Type: token • *value* comes from list: {'AgentBank'|'Lender'}
Documentation	When breakage cost is applicable, defines who is calculating it.

Diagram



Schema Component Representation

```
<xsd:simpleType name="BreakageCostEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AgentBank"/>
    <xsd:enumeration value="Lender"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

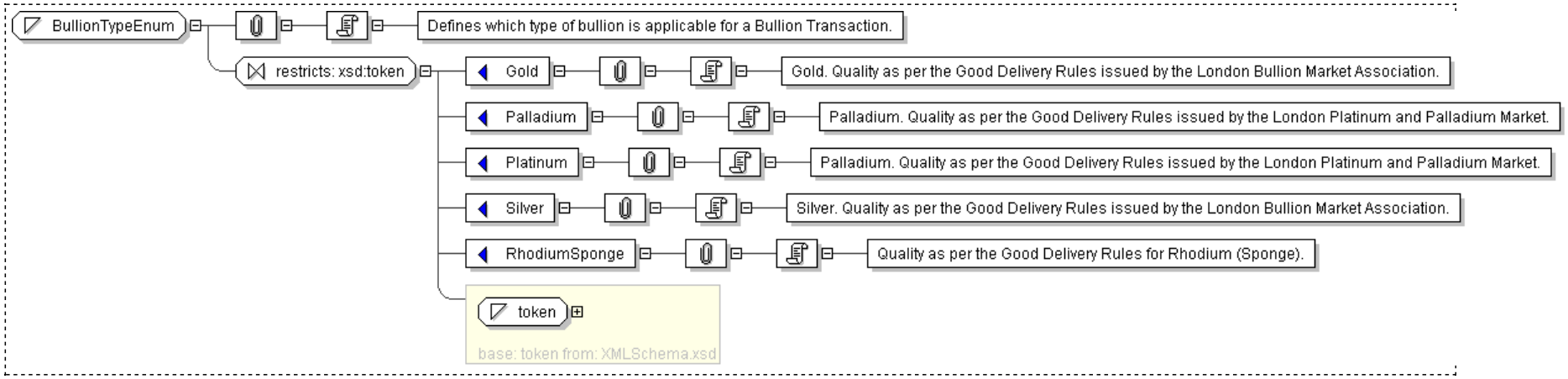
Simple Type: BullionTypeEnum

[Table of contents]

Super-types:	xsd:token < BullionTypeEnum (by restriction)
Sub-types:	None

Name	BullionTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Gold'|'Palladium'|'Platinum'|'Silver'|'RhodiumSponge'}
Documentation	Defines which type of bullion is applicable for a Bullion Transaction.

Diagram



Schema Component Representation

```
<xsd:simpleType name="BullionTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Gold"/>
    <xsd:enumeration value="Palladium"/>
    <xsd:enumeration value="Platinum"/>
    <xsd:enumeration value="Silver"/>
    <xsd:enumeration value="RhodiumSponge"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

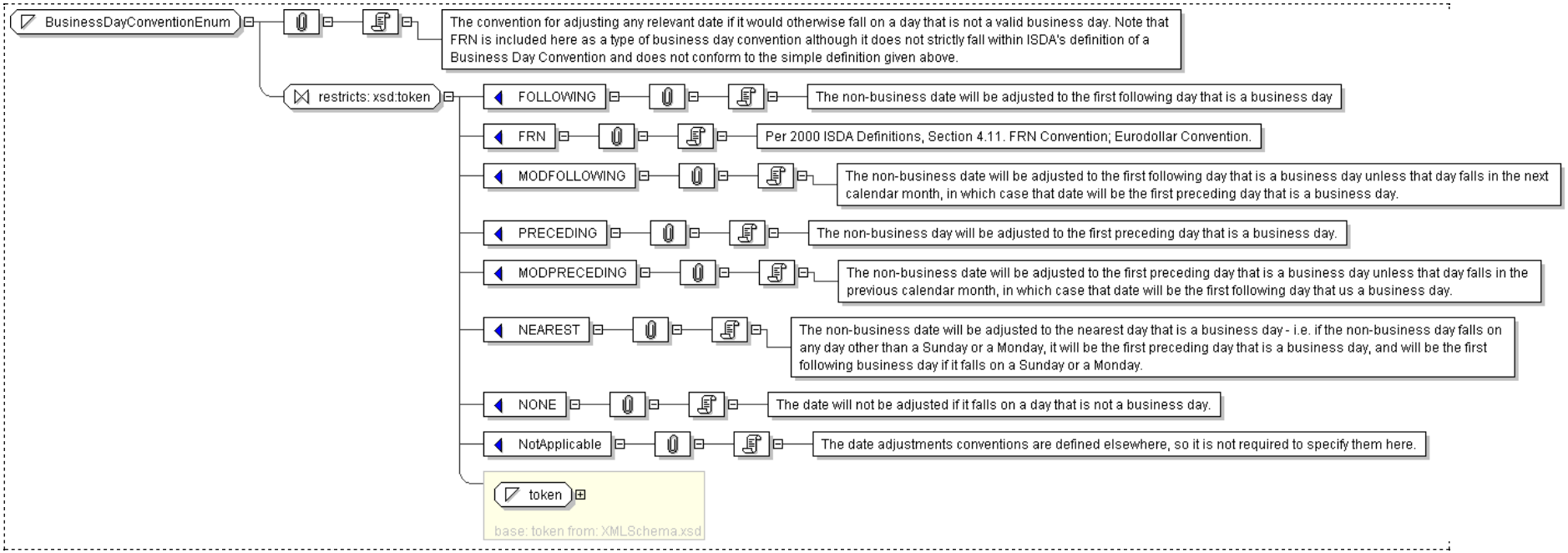
Simple Type: BusinessDayConventionEnum

[Table of contents]

Super-types:	xsd:token < BusinessDayConventionEnum (by restriction)
Sub-types:	None

Name	BusinessDayConventionEnum
Content	• Base XSD Type: token • *value* comes from list: {'FOLLOWING'|'FRN'|'MODFOLLOWING'|'PRECEDING'|'MODPRECEDING'|'NEAREST'|'NONE'|'NotApplicable'}
Documentation	The convention for adjusting any relevant date if it would otherwise fall on a day that is not a valid business day. Note that FRN is included here as a type of business day convention although it does not strictly fall within ISDA's definition of a Business Day Convention and does not conform to the simple definition given above.

Diagram



Schema Component Representation

```
<xsd:simpleType name="BusinessDayConventionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="FOLLOWING"/>
    <xsd:enumeration value="FRN"/>
    <xsd:enumeration value="MODFOLLOWING"/>
    <xsd:enumeration value="PRECEDING"/>
    <xsd:enumeration value="MODPRECEDING"/>
    <xsd:enumeration value="NEAREST"/>
    <xsd:enumeration value="NONE"/>
    <xsd:enumeration value="NotApplicable"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

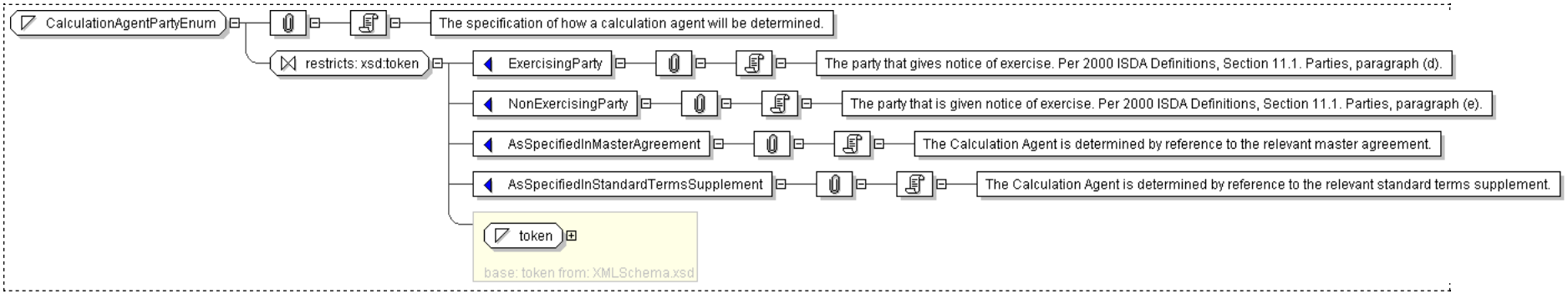
Simple Type: CalculationAgentPartyEnum

[Table of contents]

Super-types:	xsd:token < CalculationAgentPartyEnum (by restriction)
Sub-types:	None

Name	CalculationAgentPartyEnum
Content	• Base XSD Type: token • *value* comes from list: {'ExercisingParty'|'NonExercisingParty'|'AsSpecifiedInMasterAgreement'|'AsSpecifiedInStandardTermsSupplement'}
Documentation	The specification of how a calculation agent will be determined.

Diagram



Schema Component Representation

```
<xsd:simpleType name="CalculationAgentPartyEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="ExercisingParty"/>
    <xsd:enumeration value="NonExercisingParty"/>
    <xsd:enumeration value="AsSpecifiedInMasterAgreement"/>
    <xsd:enumeration value="AsSpecifiedInStandardTermsSupplement"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

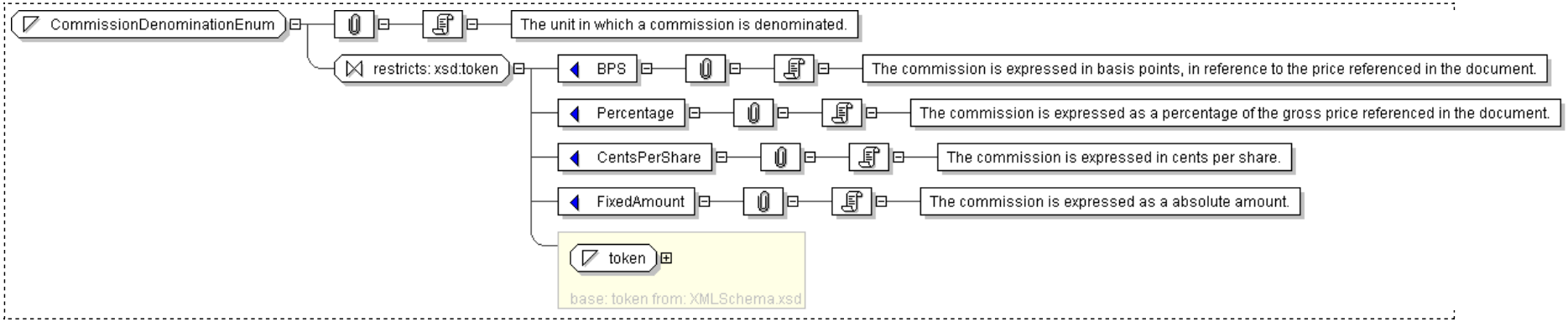
Simple Type: CommissionDenominationEnum

[Table of contents]

Super-types:	xsd:token < CommissionDenominationEnum (by restriction)
Sub-types:	None

Name	CommissionDenominationEnum
Content	• Base XSD Type: token • *value* comes from list: {'BPS'|'Percentage'|'CentsPerShare'|'FixedAmount'}
Documentation	The unit in which a commission is denominated.

Diagram



Schema Component Representation

```
<xsd:simpleType name="CommissionDenominationEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="BPS"/>
    <xsd:enumeration value="Percentage"/>
    <xsd:enumeration value="CentsPerShare"/>
    <xsd:enumeration value="FixedAmount"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

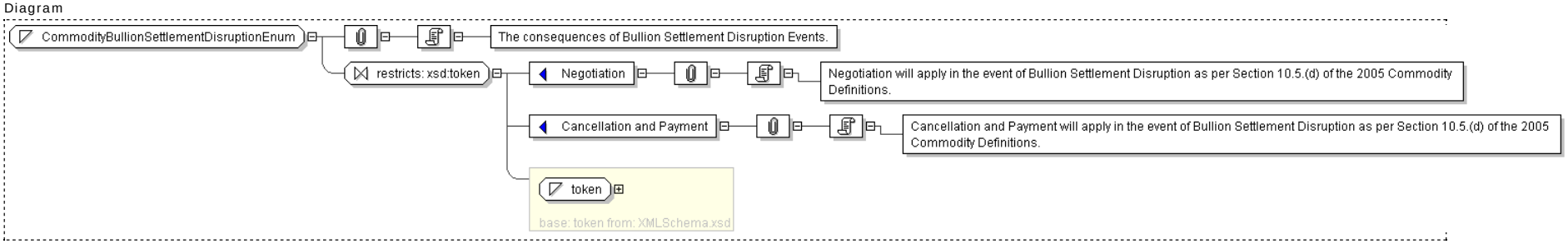
Simple Type: CommodityBullionSettlementDisruptionEnum

[Table of contents]

Super-types:	xsd:token < CommodityBullionSettlementDisruptionEnum (by restriction)
Sub-types:	None

Name	CommodityBullionSettlementDisruptionEnum
Content	• Base XSD Type: token • *value* comes from list: {'Negotiation'|'Cancellation and Payment'}
Documentation	The consequences of Bullion Settlement Disruption Events.

Diagram



Schema Component Representation

```
<xsd:simpleType name="CommodityBullionSettlementDisruptionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Negotiation"/>
    <xsd:enumeration value="Cancellation and Payment"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Simple Type: CommodityDayTypeEnum

[Table of contents]

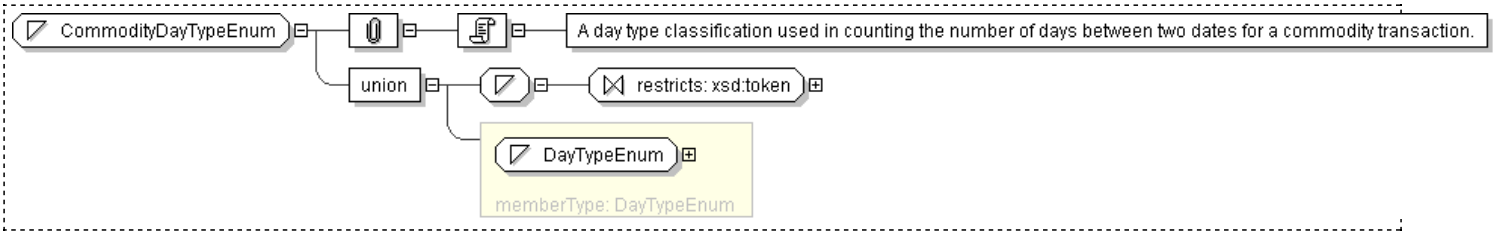
Super-types:	None
Sub-types:	None

Name	CommodityDayTypeEnum

Content	• Union of following types:
	  ◦ DayTypeEnum
	  ◦ Locally defined type:
	  ▪ Base XSD Type: token
	  ▪ *value* comes from list: {'GasFlow'}

Documentation	A day type classification used in counting the number of days between two dates for a commodity transaction.

Diagram



Schema Component Representation

```
<xsd:simpleType name="CommodityDayTypeEnum">
  <xsd:union memberTypes=" DayTypeEnum ">
    <xsd:simpleType>
      <xsd:restriction base=" xsd:token ">
        <xsd:enumeration value="GasFlow"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

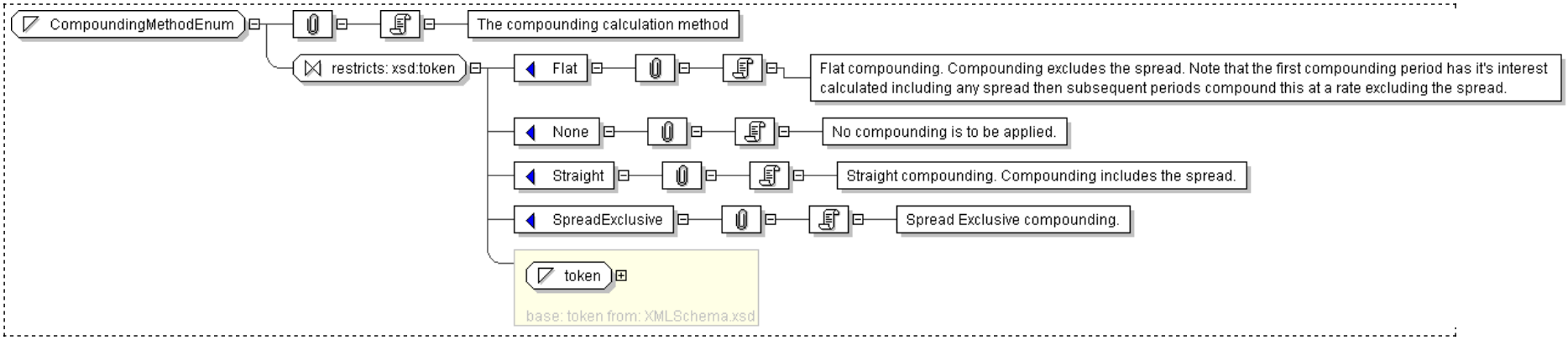
Simple Type: CompoundingMethodEnum

[Table of contents]

Super-types:	xsd:token < CompoundingMethodEnum (by restriction)
Sub-types:	None

Name	CompoundingMethodEnum
Content	• Base XSD Type: token • *value* comes from list: {'Flat'|'None'|'Straight'|'SpreadExclusive'}
Documentation	The compounding calculation method

Diagram



Schema Component Representation

```
<xsd:simpleType name="CompoundingMethodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Flat"/>
    <xsd:enumeration value="None"/>
    <xsd:enumeration value="Straight"/>
    <xsd:enumeration value="SpreadExclusive"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

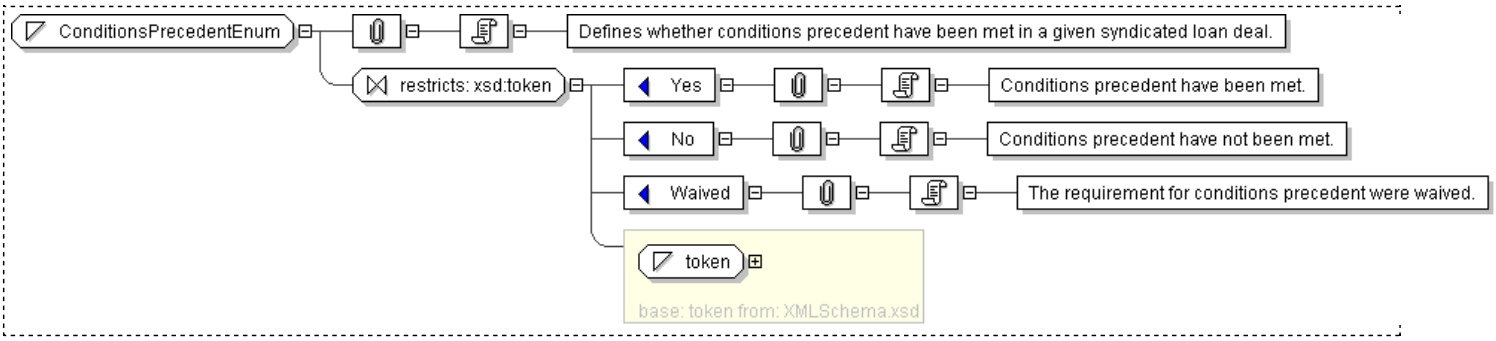
Simple Type: ConditionsPrecedentEnum

[Table of contents]

Super-types:	xsd:token < ConditionsPrecedentEnum (by restriction)
Sub-types:	None

Name	ConditionsPrecedentEnum
Content	• Base XSD Type: token • *value* comes from list: {'Yes'|'No'|'Waived'}
Documentation	Defines whether conditions precedent have been met in a given syndicated loan deal.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ConditionsPrecedentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Yes"/>
    <xsd:enumeration value="No"/>
    <xsd:enumeration value="Waived"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

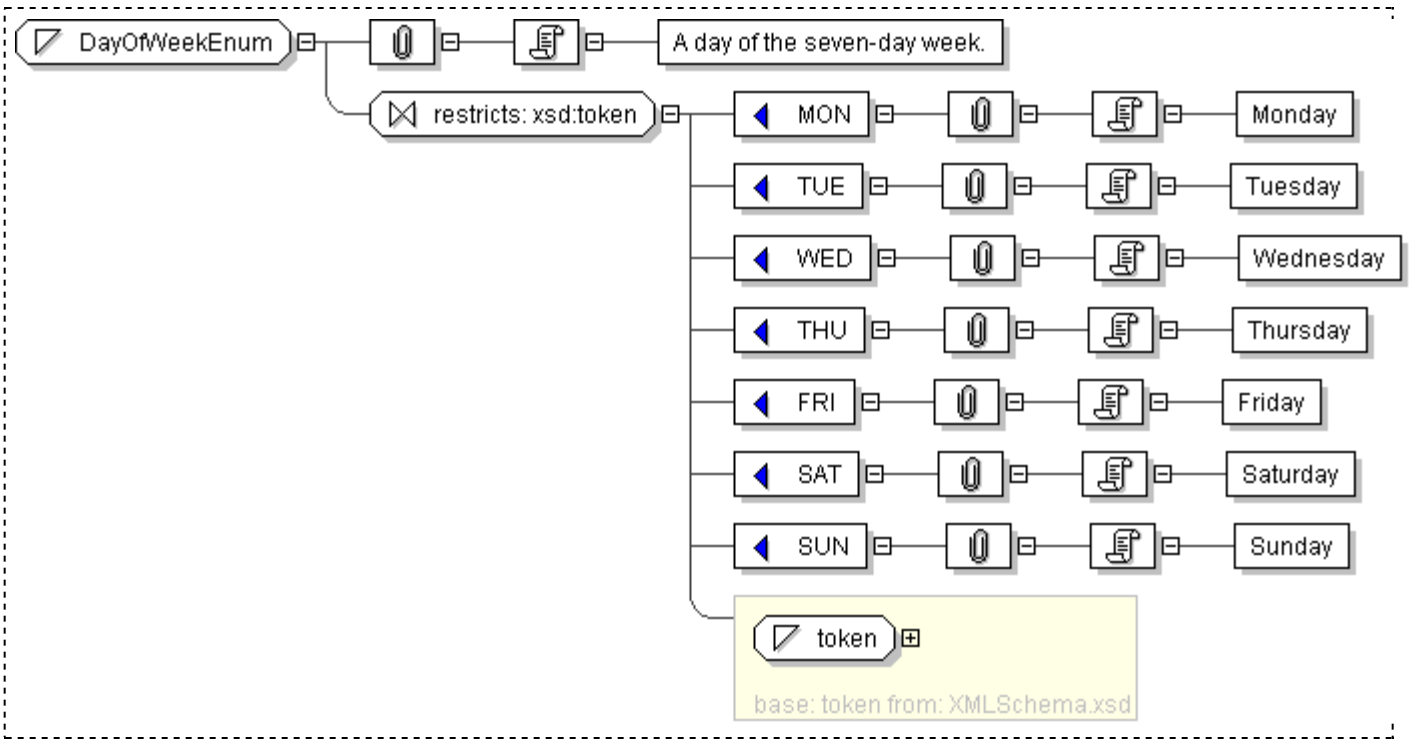
Simple Type: DayOfWeekEnum

[Table of contents]

Super-types:	xsd:token < DayOfWeekEnum (by restriction)
Sub-types:	None

Name	DayOfWeekEnum
Content	• Base XSD Type: token • *value* comes from list: {'MON'|'TUE'|'WED'|'THU'|'FRI'|'SAT'|'SUN'}
Documentation	A day of the seven-day week.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DayOfWeekEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="MON"/>
      <xsd:enumeration value="TUE"/>
      <xsd:enumeration value="WED"/>
      <xsd:enumeration value="THU"/>
      <xsd:enumeration value="FRI"/>
      <xsd:enumeration value="SAT"/>
      <xsd:enumeration value="SUN"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

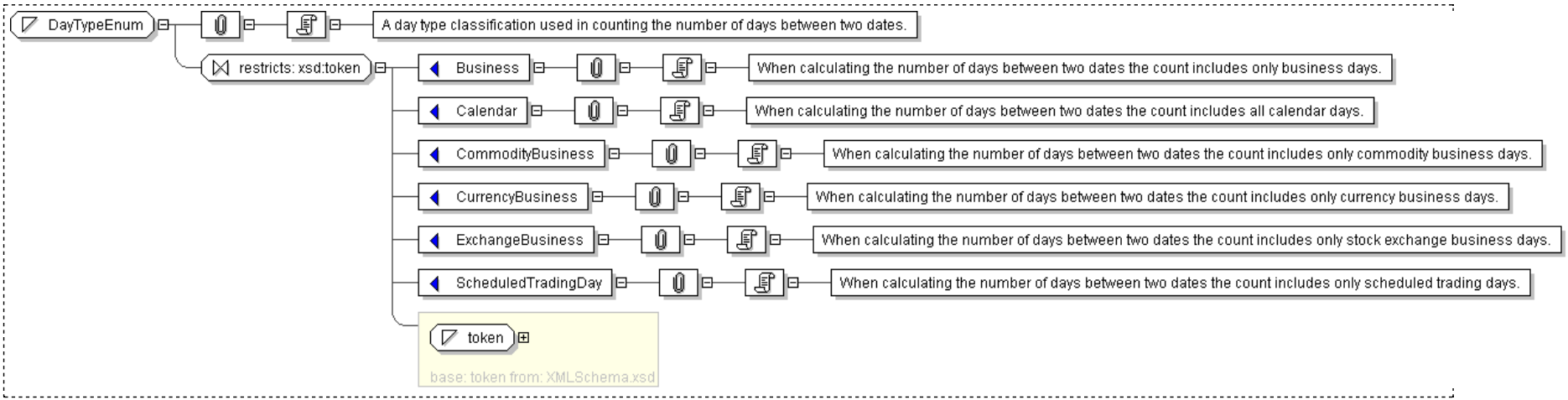
Simple Type: DayTypeEnum

[Table of contents]

Super-types:	xsd:token < DayTypeEnum (by restriction)
Sub-types:	None

Name	DayTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Business'|'Calendar'|'CommodityBusiness'|'CurrencyBusiness'|'ExchangeBusiness'|'ScheduledTradingDay'}
Documentation	A day type classification used in counting the number of days between two dates.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DayTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Business"/>
    <xsd:enumeration value="Calendar"/>
    <xsd:enumeration value="CommodityBusiness"/>
    <xsd:enumeration value="CurrencyBusiness"/>
    <xsd:enumeration value="ExchangeBusiness"/>
    <xsd:enumeration value="ScheduledTradingDay"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

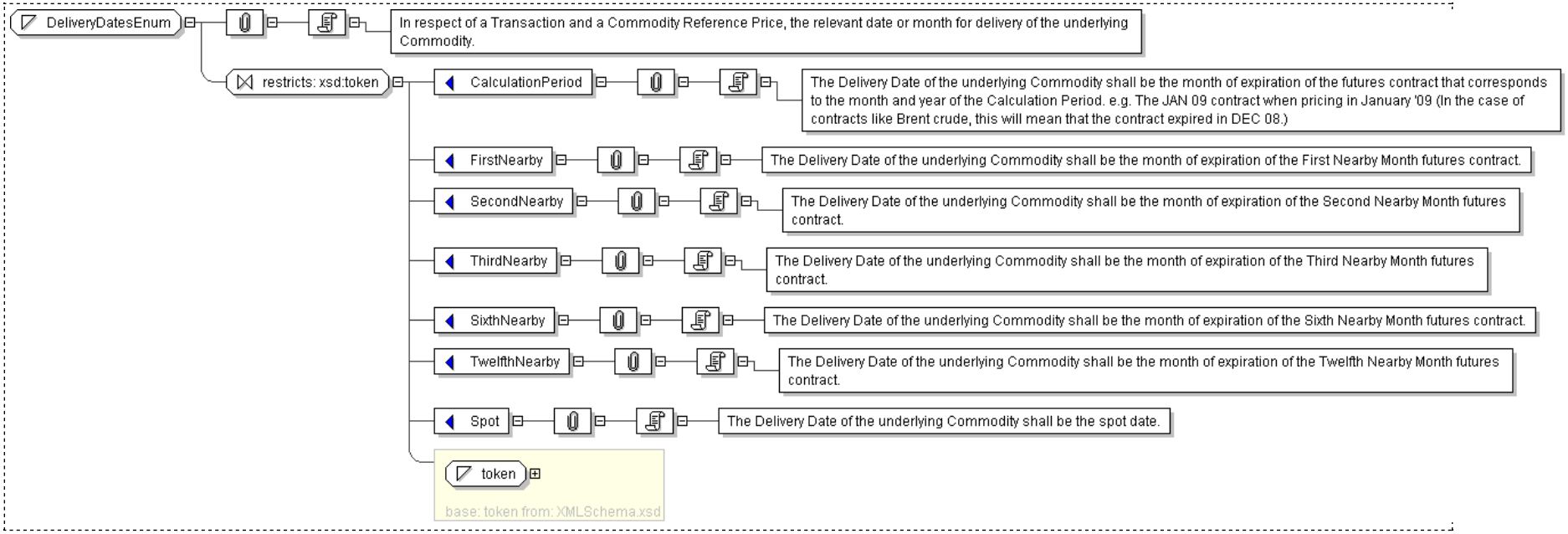
Simple Type: DeliveryDatesEnum

[Table of contents]

Super-types:	xsd:token < DeliveryDatesEnum (by restriction)
Sub-types:	None

Name	DeliveryDatesEnum
Content	• Base XSD Type: token • *value* comes from list: {'CalculationPeriod'|'FirstNearby'|'SecondNearby'|'ThirdNearby'|'SixthNearby'|'TwelfthNearby'|'Spot'}
Documentation	In respect of a Transaction and a Commodity Reference Price, the relevant date or month for delivery of the underlying Commodity.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DeliveryDatesEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CalculationPeriod"/>
    <xsd:enumeration value="FirstNearby"/>
    <xsd:enumeration value="SecondNearby"/>
    <xsd:enumeration value="ThirdNearby"/>
    <xsd:enumeration value="SixthNearby"/>
    <xsd:enumeration value="TwelfthNearby"/>
    <xsd:enumeration value="Spot"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

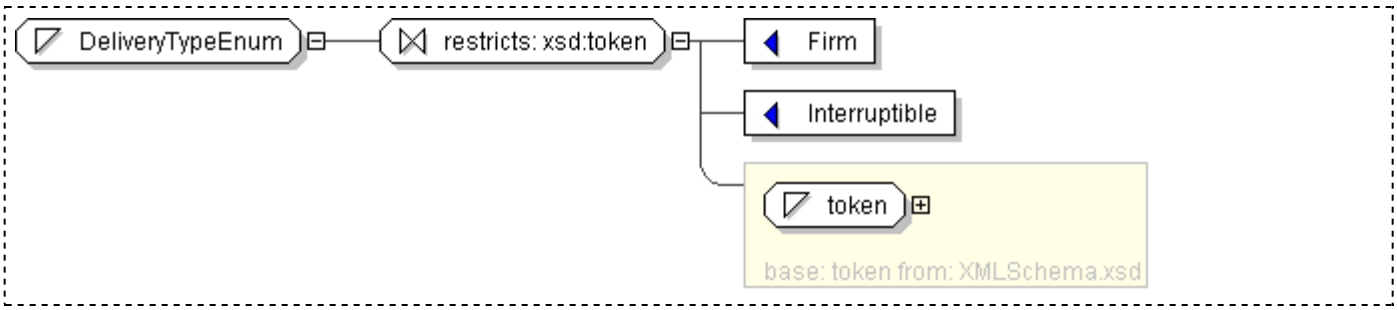
Simple Type: DeliveryTypeEnum

[Table of contents]

Super-types:	xsd:token < DeliveryTypeEnum (by restriction)
Sub-types:	None

Name	DeliveryTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Firm'|'Interruptible'}

Diagram



Schema Component Representation

```
<xsd:simpleType name="DeliveryTypeEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="Firm"/>
      <xsd:enumeration value="Interruptible"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

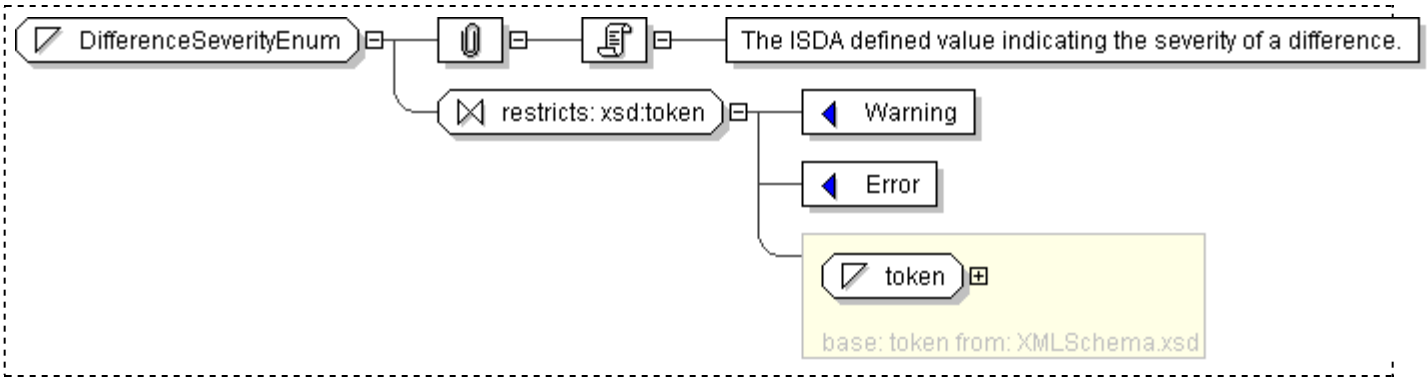
Simple Type: DifferenceSeverityEnum

[Table of contents]

Super-types:	xsd:token < DifferenceSeverityEnum (by restriction)
Sub-types:	None

Name	DifferenceSeverityEnum
Content	• Base XSD Type: token • *value* comes from list: {'Warning'|'Error'}
Documentation	The ISDA defined value indicating the severity of a difference.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DifferenceSeverityEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="Warning"/>
      <xsd:enumeration value="Error"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

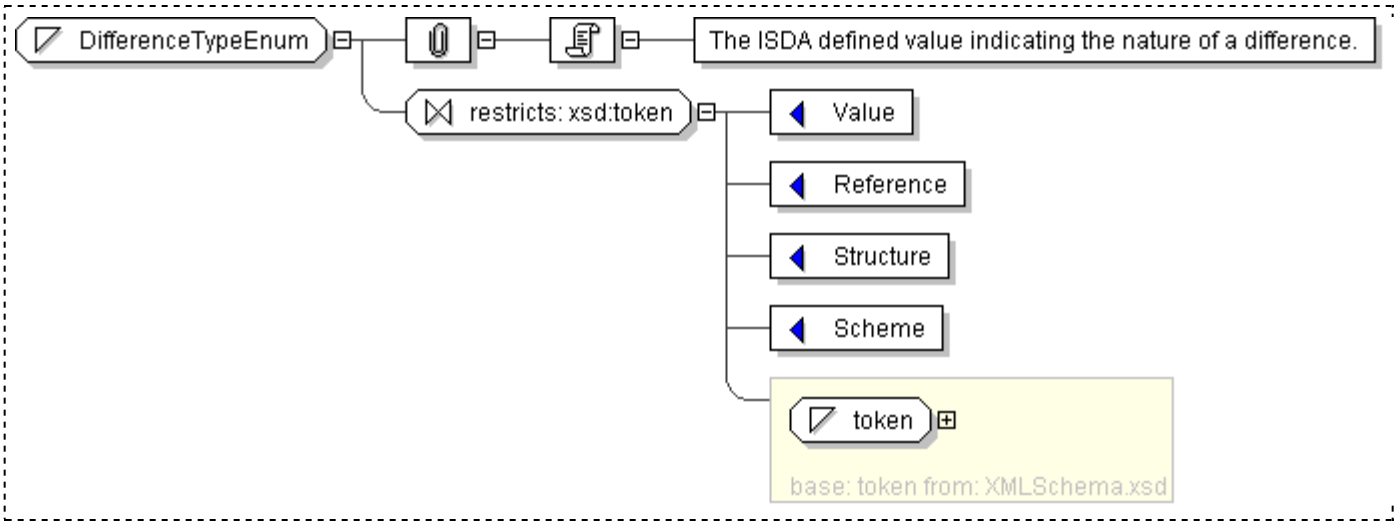
Simple Type: DifferenceTypeEnum

[Table of contents]

Super-types:	xsd:token < DifferenceTypeEnum (by restriction)
Sub-types:	None

Name	DifferenceTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Value'|'Reference'|'Structure'|'Scheme'}
Documentation	The ISDA defined value indicating the nature of a difference.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DifferenceTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Value"/>
    <xsd:enumeration value="Reference"/>
    <xsd:enumeration value="Structure"/>
    <xsd:enumeration value="Scheme"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

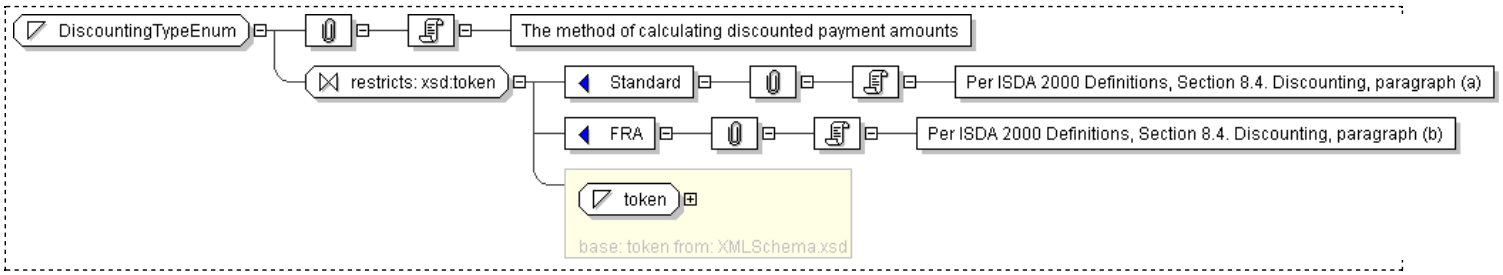
Simple Type: DiscountingTypeEnum

[Table of contents]

Super-types:	xsd:token < DiscountingTypeEnum (by restriction)
Sub-types:	None

Name	DiscountingTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Standard'|'FRA'}
Documentation	The method of calculating discounted payment amounts

Diagram



Schema Component Representation

```
<xsd:simpleType name="DiscountingTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Standard"/>
    <xsd:enumeration value="FRA"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

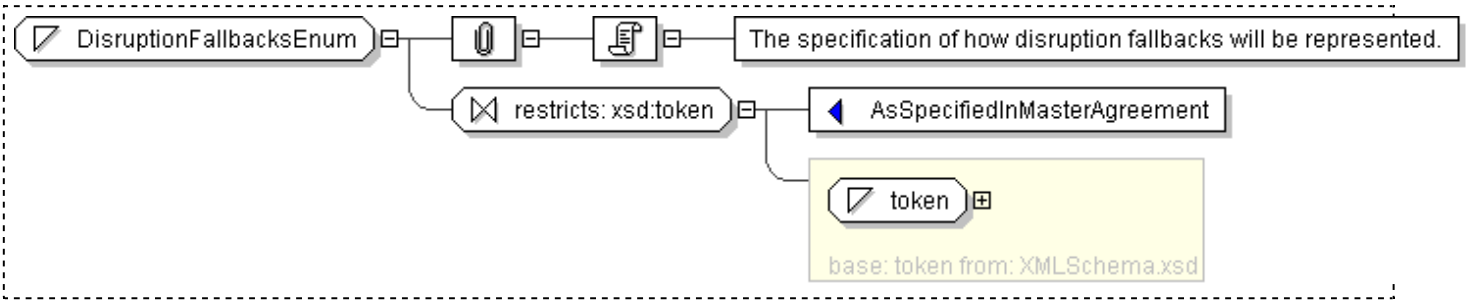
Simple Type: DisruptionFallbacksEnum

[Table of contents]

Super-types:	xsd:token < DisruptionFallbacksEnum (by restriction)
Sub-types:	None

Name	DisruptionFallbacksEnum
Content	• Base XSD Type: token • *value* comes from list: {'AsSpecifiedInMasterAgreement'}
Documentation	The specification of how disruption fallbacks will be represented.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DisruptionFallbacksEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AsSpecifiedInMasterAgreement"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

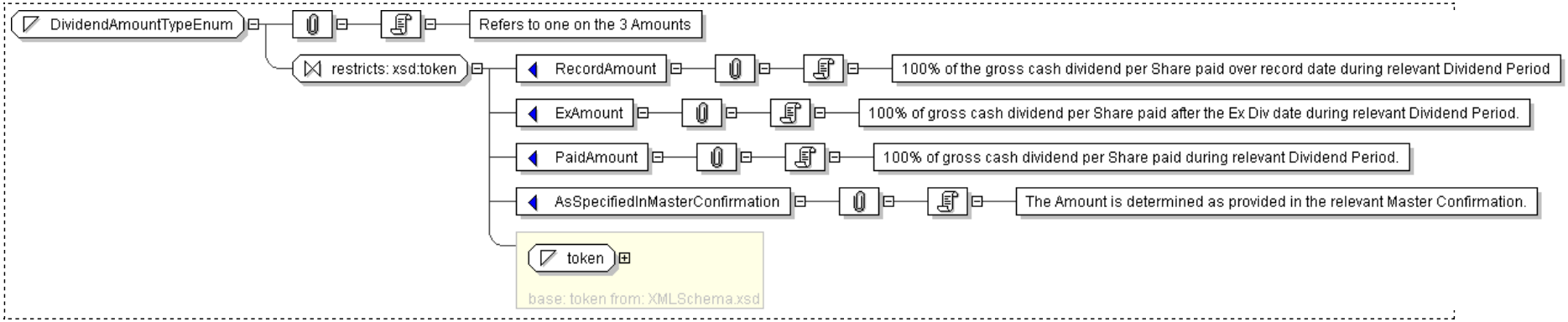
Simple Type: DividendAmountTypeEnum

[Table of contents]

Super-types:	xsd:token < DividendAmountTypeEnum (by restriction)
Sub-types:	None

Name	DividendAmountTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'RecordAmount'|'ExAmount'|'PaidAmount'|'AsSpecifiedInMasterConfirmation'}
Documentation	Refers to one on the 3 Amounts

Diagram



Schema Component Representation

```
<xsd:simpleType name="DividendAmountTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="RecordAmount"/>
    <xsd:enumeration value="ExAmount"/>
    <xsd:enumeration value="PaidAmount"/>
    <xsd:enumeration value="AsSpecifiedInMasterConfirmation"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

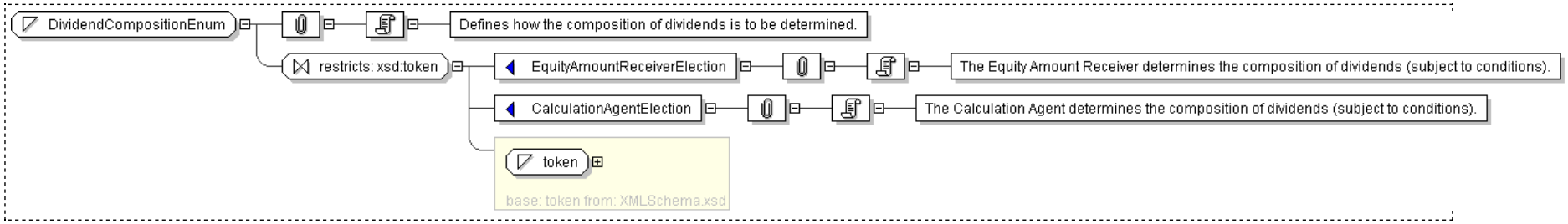
Simple Type: DividendCompositionEnum

[Table of contents]

Super-types:	xsd:token < DividendCompositionEnum (by restriction)
Sub-types:	None

Name	DividendCompositionEnum
Content	• Base XSD Type: token • *value* comes from list: {'EquityAmountReceiverElection'|'CalculationAgentElection'}
Documentation	Defines how the composition of dividends is to be determined.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DividendCompositionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="EquityAmountReceiverElection"/>
    <xsd:enumeration value="CalculationAgentElection"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

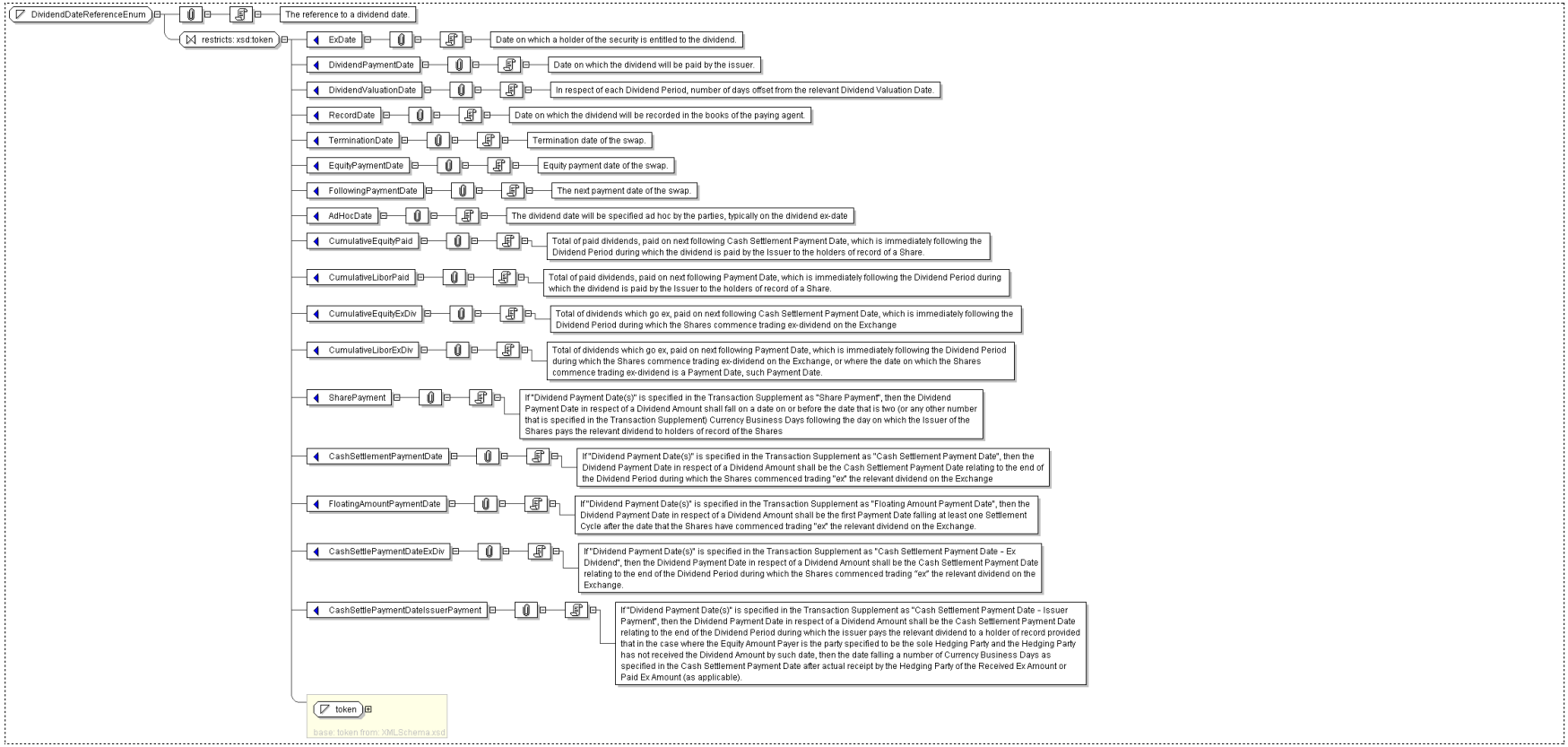
Simple Type: DividendDateReferenceEnum

[Table of contents]

Super-types:	xsd:token < DividendDateReferenceEnum (by restriction)
Sub-types:	None

Name	DividendDateReferenceEnum
Content	• Base XSD Type: token • *value* comes from list: {'ExDate'|'DividendPaymentDate'|'DividendValuationDate'|'RecordDate'|'TerminationDate'|'EquityPaymentDate'|'FollowingPaymentDate'|'AdHocDate'|'CumulativeEquityPaid'|'CumulativeLiborPaid'|'CumulativeEquityExDiv'|'CumulativeLiborExDiv'|'SharePayment'|'CashSettlementPaymentDate'|'FloatingAmountPaymentDate'|'CashSettlePaymentDateExDiv'|'CashSettlePaymentDateIssuerPayment'}
Documentation	The reference to a dividend date.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DividendDateReferenceEnum">
  <xsd:restriction base=" xsd:token ">
     <xsd:enumeration value="ExDate"/>
     <xsd:enumeration value="DividendPaymentDate"/>
     <xsd:enumeration value="DividendValuationDate"/>
     <xsd:enumeration value="RecordDate"/>
     <xsd:enumeration value="TerminationDate"/>
     <xsd:enumeration value="EquityPaymentDate"/>
     <xsd:enumeration value="FollowingPaymentDate"/>
     <xsd:enumeration value="AdHocDate"/>
     <xsd:enumeration value="CumulativeEquityPaid"/>
     <xsd:enumeration value="CumulativeLiborPaid"/>
     <xsd:enumeration value="CumulativeEquityExDiv"/>
     <xsd:enumeration value="CumulativeLiborExDiv"/>
     <xsd:enumeration value="SharePayment"/>
     <xsd:enumeration value="CashSettlementPaymentDate"/>
     <xsd:enumeration value="FloatingAmountPaymentDate"/>
     <xsd:enumeration value="CashSettlePaymentDateExDiv"/>
     <xsd:enumeration value="CashSettlePaymentDateIssuerPayment"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

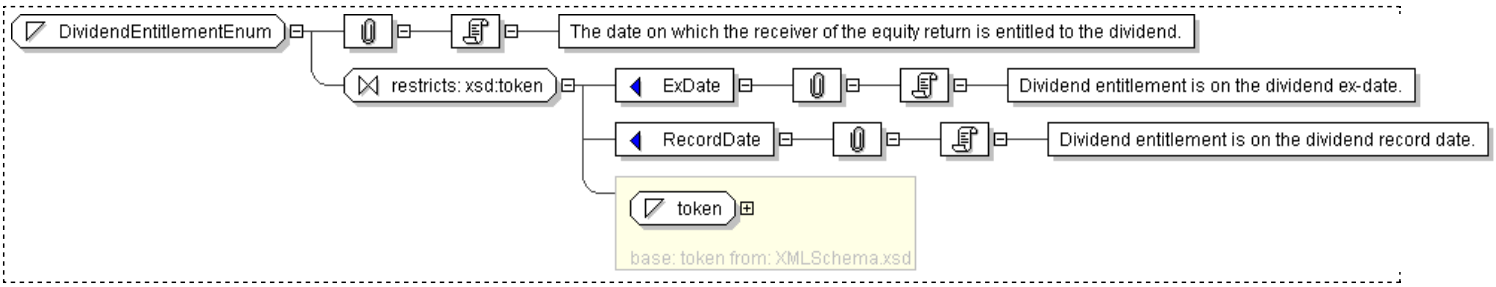
Simple Type: DividendEntitlementEnum

[Table of contents]

Super-types:	xsd:token < DividendEntitlementEnum (by restriction)
Sub-types:	None

Name	DividendEntitlementEnum
Content	• Base XSD Type: token • *value* comes from list: {'ExDate'|'RecordDate'}
Documentation	The date on which the receiver of the equity return is entitled to the dividend.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DividendEntitlementEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="ExDate"/>
    <xsd:enumeration value="RecordDate"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

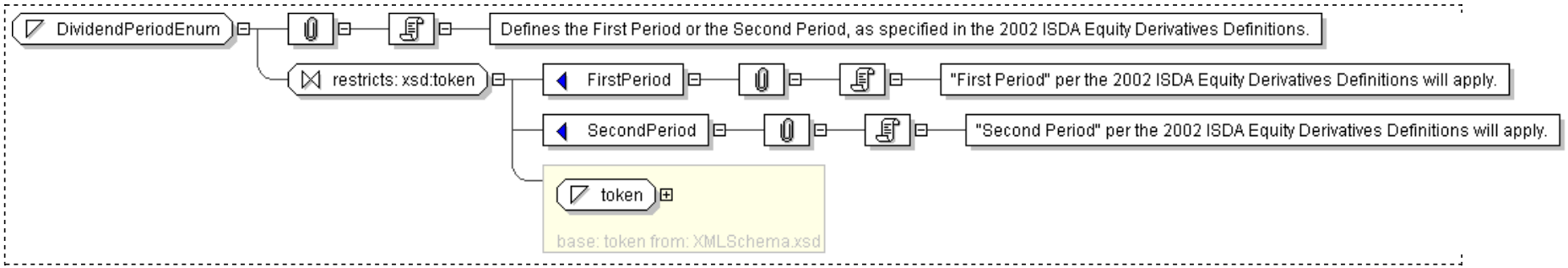
Simple Type: DividendPeriodEnum

[Table of contents]

Super-types:	xsd:token < DividendPeriodEnum (by restriction)
Sub-types:	None

Name	DividendPeriodEnum
Content	• Base XSD Type: token • *value* comes from list: {'FirstPeriod'|'SecondPeriod'}
Documentation	Defines the First Period or the Second Period, as specified in the 2002 ISDA Equity Derivatives Definitions.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DividendPeriodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="FirstPeriod"/>
    <xsd:enumeration value="SecondPeriod"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

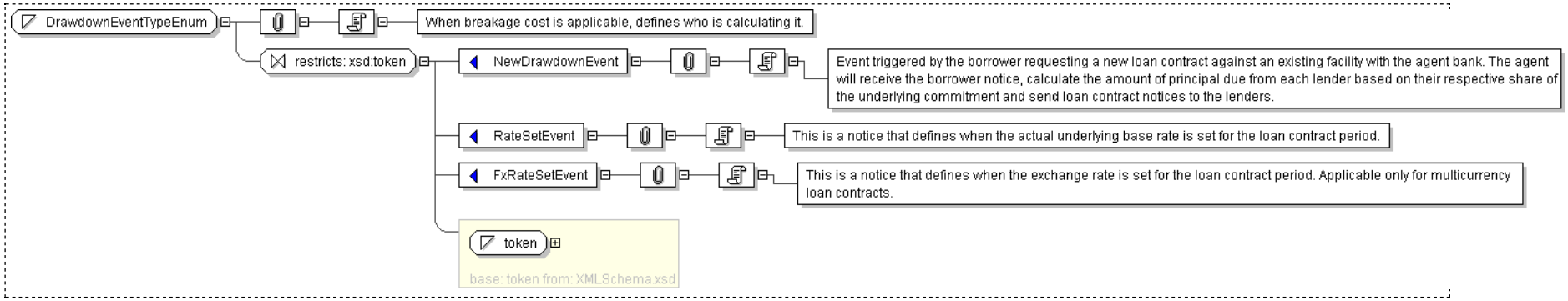
Simple Type: DrawdownEventTypeEnum

[Table of contents]

Super-types:	xsd:token < DrawdownEventTypeEnum (by restriction)
Sub-types:	None

Name	DrawdownEventTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'NewDrawdownEvent'|'RateSetEvent'|'FxRateSetEvent'}
Documentation	When breakage cost is applicable, defines who is calculating it.

Diagram



Schema Component Representation

```
<xsd:simpleType name="DrawdownEventTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="NewDrawdownEvent"/>
    <xsd:enumeration value="RateSetEvent"/>
    <xsd:enumeration value="FxRateSetEvent"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

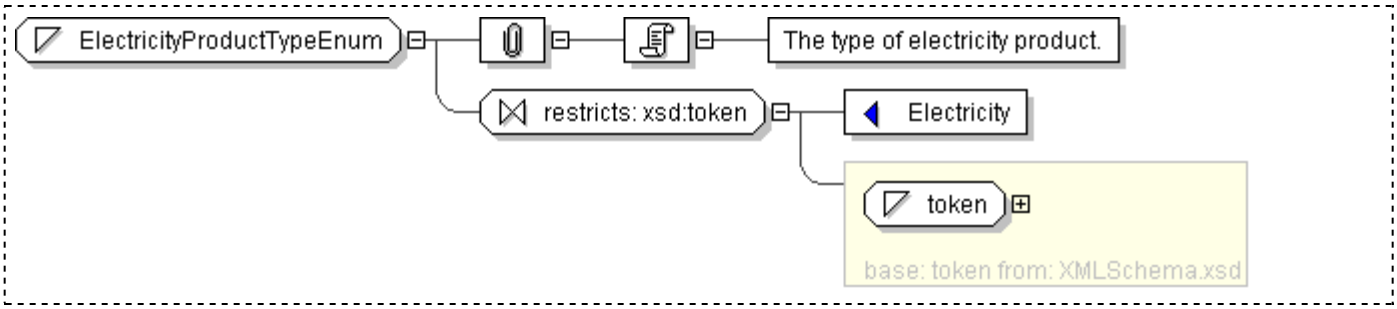
Simple Type: ElectricityProductTypeEnum

[Table of contents]

Super-types:	xsd:token < ElectricityProductTypeEnum (by restriction)
Sub-types:	None

Name	ElectricityProductTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Electricity'}
Documentation	The type of electricity product.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ElectricityProductTypeEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="Electricity"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

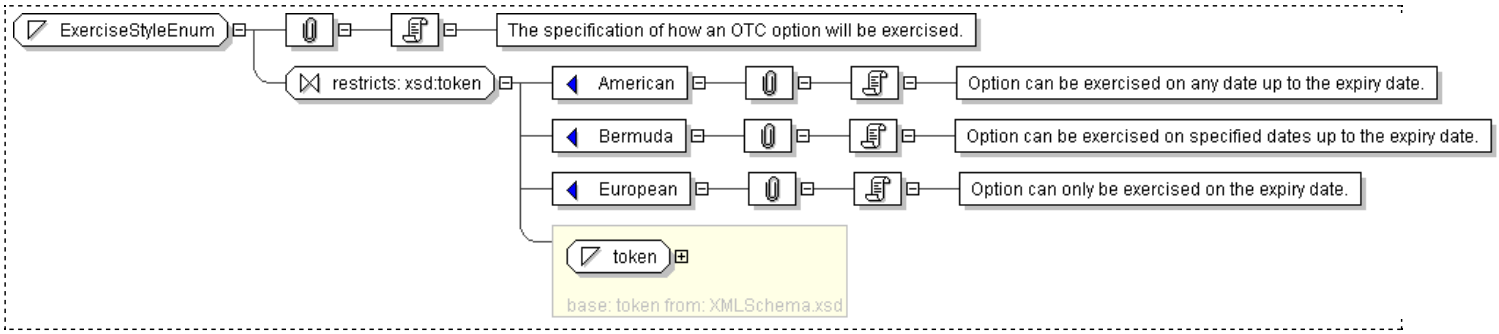
Simple Type: ExerciseStyleEnum

[Table of contents]

Super-types:	xsd:token < ExerciseStyleEnum (by restriction)
Sub-types:	None

Name	ExerciseStyleEnum
Content	• Base XSD Type: token • *value* comes from list: {'American'|'Bermuda'|'European'}
Documentation	The specification of how an OTC option will be exercised.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ExerciseStyleEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="American"/>
    <xsd:enumeration value="Bermuda"/>
    <xsd:enumeration value="European"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

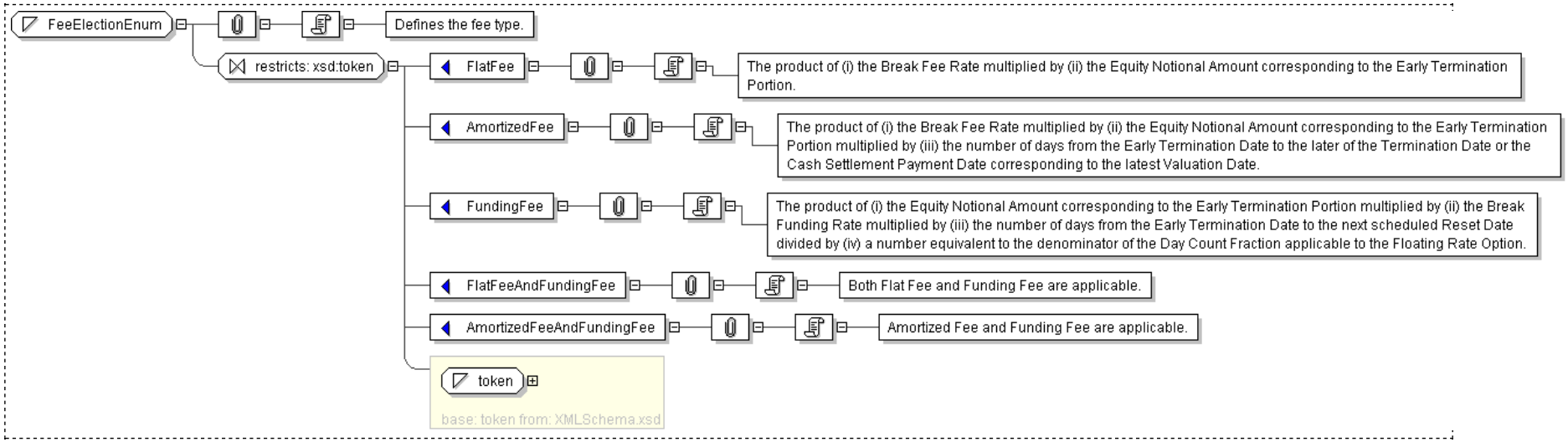
Simple Type: FeeElectionEnum

[Table of contents]

Super-types:	xsd:token < FeeElectionEnum (by restriction)
Sub-types:	None

Name	FeeElectionEnum
Content	• Base XSD Type: token • *value* comes from list: {'FlatFee'|'AmortizedFee'|'FundingFee'|'FlatFeeAndFundingFee'|'AmortizedFeeAndFundingFee'}
Documentation	Defines the fee type.

Diagram



Schema Component Representation

```
<xsd:simpleType name="FeeElectionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="FlatFee"/>
    <xsd:enumeration value="AmortizedFee"/>
    <xsd:enumeration value="FundingFee"/>
    <xsd:enumeration value="FlatFeeAndFundingFee"/>
    <xsd:enumeration value="AmortizedFeeAndFundingFee"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

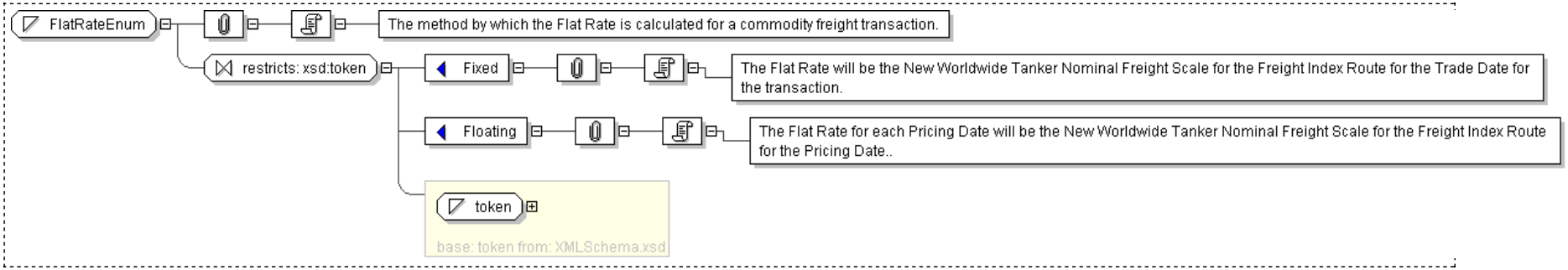
Simple Type: FlatRateEnum

[Table of contents]

Super-types:	xsd:token < FlatRateEnum (by restriction)
Sub-types:	None

Name	FlatRateEnum
Content	• Base XSD Type: token • *value* comes from list: {'Fixed'|'Floating'}
Documentation	The method by which the Flat Rate is calculated for a commodity freight transaction.

Diagram



Schema Component Representation

```
<xsd:simpleType name="FlatRateEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Fixed"/>
    <xsd:enumeration value="Floating"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

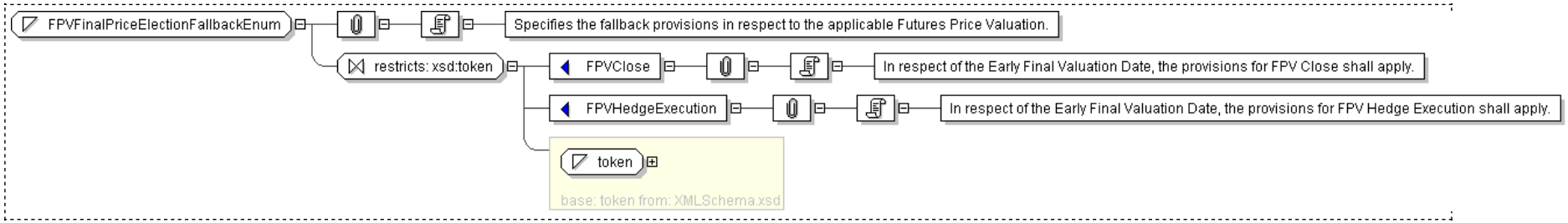
Simple Type: FPVFinalPriceElectionFallbackEnum

[Table of contents]

Super-types:	xsd:token < FPVFinalPriceElectionFallbackEnum (by restriction)
Sub-types:	None

Name	FPVFinalPriceElectionFallbackEnum
Content	• Base XSD Type: token • *value* comes from list: {'FPVClose'|'FPVHedgeExecution'}
Documentation	Specifies the fallback provisions in respect to the applicable Futures Price Valuation.

Diagram



Schema Component Representation

```
<xsd:simpleType name="FPVFinalPriceElectionFallbackEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="FPVClose"/>
    <xsd:enumeration value="FPVHedgeExecution"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

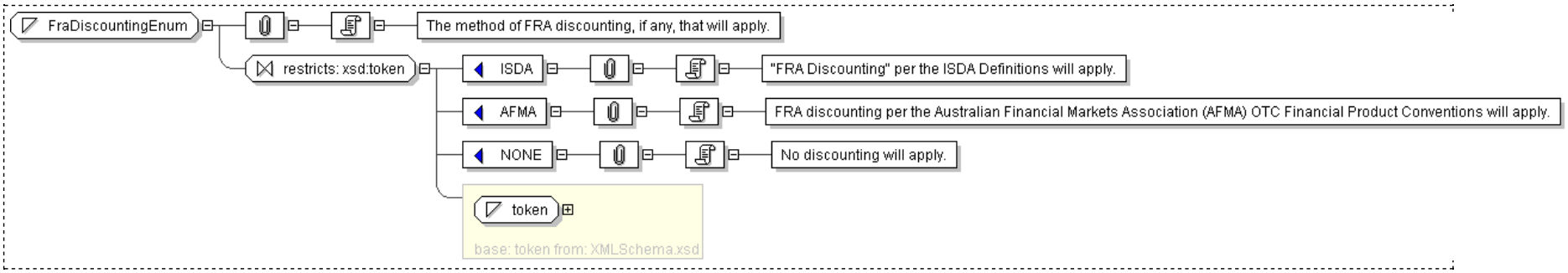
Simple Type: FraDiscountingEnum

[Table of contents]

Super-types:	xsd:token < FraDiscountingEnum (by restriction)
Sub-types:	None

Name	FraDiscountingEnum
Content	• Base XSD Type: token • *value* comes from list: {'ISDA'|'AFMA'|'NONE'}
Documentation	The method of FRA discounting, if any, that will apply.

Diagram



Schema Component Representation

```
<xsd:simpleType name="FraDiscountingEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="ISDA"/>
    <xsd:enumeration value="AFMA"/>
    <xsd:enumeration value="NONE"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

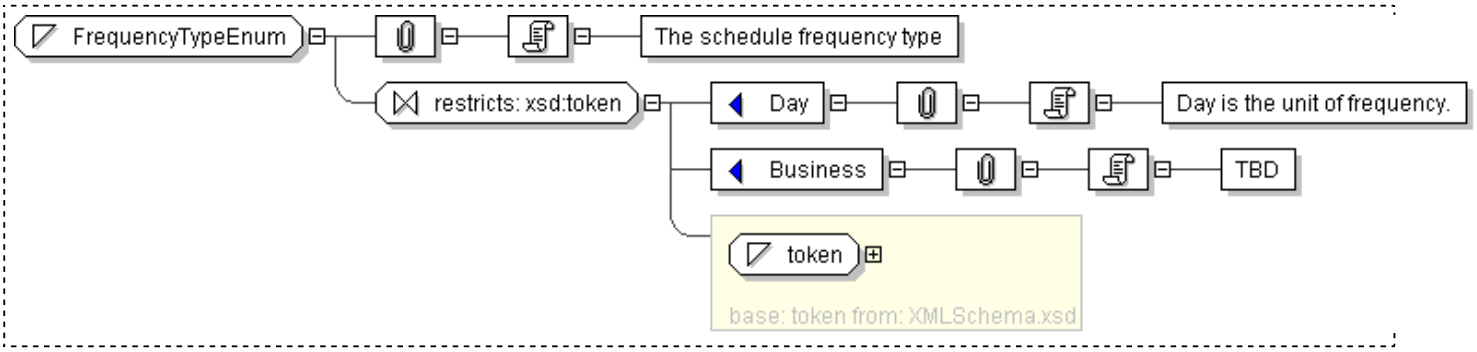
Simple Type: FrequencyTypeEnum

[Table of contents]

Super-types:	xsd:token < FrequencyTypeEnum (by restriction)
Sub-types:	None

Name	FrequencyTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Day'|'Business'}
Documentation	The schedule frequency type

Diagram



Schema Component Representation

```
<xsd:simpleType name="FrequencyTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Day"/>
    <xsd:enumeration value="Business"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

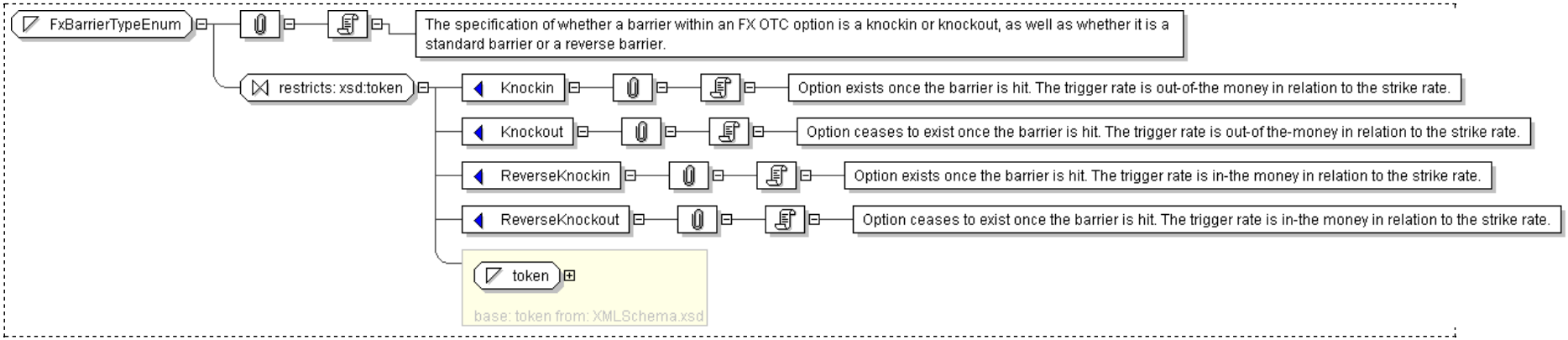
Simple Type: FxBarrierTypeEnum

[Table of contents]

Super-types:	xsd:token < FxBarrierTypeEnum (by restriction)
Sub-types:	None

Name	FxBarrierTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Knockin'|'Knockout'|'ReverseKnockin'|'ReverseKnockout'}
Documentation	The specification of whether a barrier within an FX OTC option is a knockin or knockout, as well as whether it is a standard barrier or a reverse barrier.

Diagram



Schema Component Representation

```
<xsd:simpleType name="FxBarrierTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Knockin"/>
    <xsd:enumeration value="Knockout"/>
    <xsd:enumeration value="ReverseKnockin"/>
    <xsd:enumeration value="ReverseKnockout"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

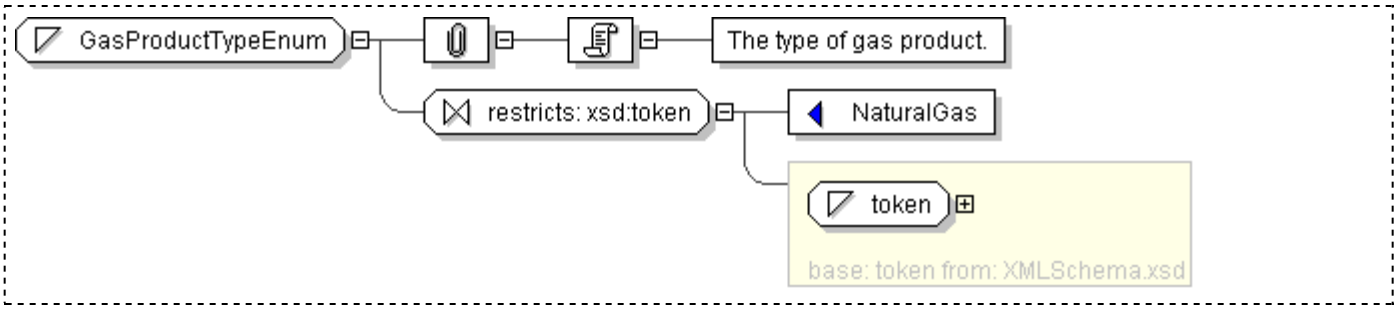
Simple Type: GasProductTypeEnum

[Table of contents]

Super-types:	xsd:token < GasProductTypeEnum (by restriction)
Sub-types:	None

Name	GasProductTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'NaturalGas'}
Documentation	The type of gas product.

Diagram



Schema Component Representation

```
<xsd:simpleType name="GasProductTypeEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="NaturalGas"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

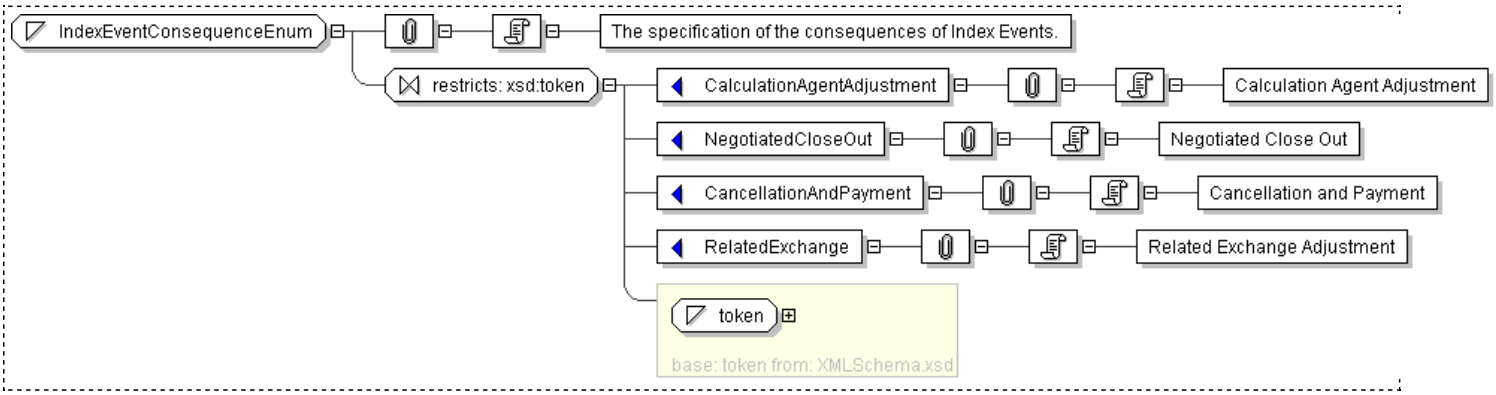
Simple Type: IndexEventConsequenceEnum

[Table of contents]

Super-types:	xsd:token < IndexEventConsequenceEnum (by restriction)
Sub-types:	None

Name	IndexEventConsequenceEnum
Content	• Base XSD Type: token • *value* comes from list: {'CalculationAgentAdjustment'|'NegotiatedCloseOut'|'CancellationAndPayment'|'RelatedExchange'}
Documentation	The specification of the consequences of Index Events.

Diagram



Schema Component Representation

```
<xsd:simpleType name="IndexEventConsequenceEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CalculationAgentAdjustment"/>
    <xsd:enumeration value="NegotiatedCloseOut"/>
    <xsd:enumeration value="CancellationAndPayment"/>
    <xsd:enumeration value="RelatedExchange"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

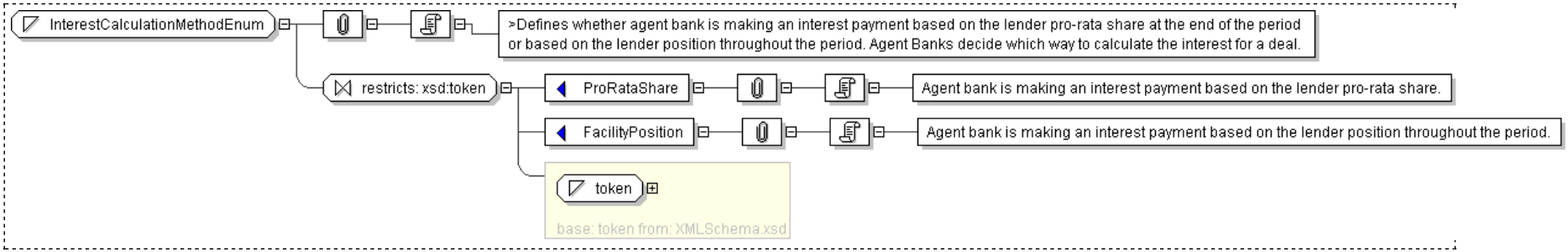
Simple Type: InterestCalculationMethodEnum

[Table of contents]

Super-types:	xsd:token < InterestCalculationMethodEnum (by restriction)
Sub-types:	None

Name	InterestCalculationMethodEnum
Content	• Base XSD Type: token • *value* comes from list: {'ProRataShare'|'FacilityPosition'}
Documentation	>Defines whether agent bank is making an interest payment based on the lender pro-rata share at the end of the period or based on the lender position throughout the period. Agent Banks decide which way to calculate the interest for a deal.

Diagram



Schema Component Representation

```
<xsd:simpleType name="InterestCalculationMethodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="ProRataShare"/>
    <xsd:enumeration value="FacilityPosition"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

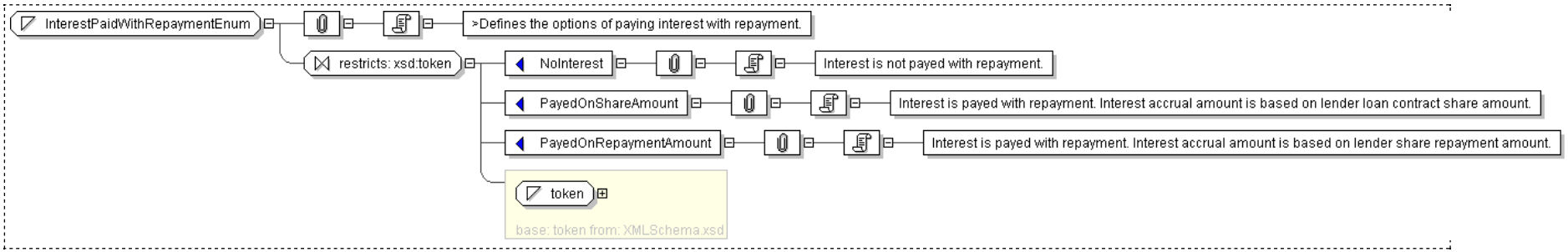
Simple Type: InterestPaidWithRepaymentEnum

[Table of contents]

Super-types:	xsd:token < InterestPaidWithRepaymentEnum (by restriction)
Sub-types:	None

Name	InterestPaidWithRepaymentEnum
Content	• Base XSD Type: token • *value* comes from list: {'NoInterest'|'PayedOnShareAmount'|'PayedOnRepaymentAmount'}
Documentation	>Defines the options of paying interest with repayment.

Diagram



Schema Component Representation

```
<xsd:simpleType name="InterestPaidWithRepaymentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="NoInterest"/>
    <xsd:enumeration value="PayedOnShareAmount"/>
    <xsd:enumeration value="PayedOnRepaymentAmount"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

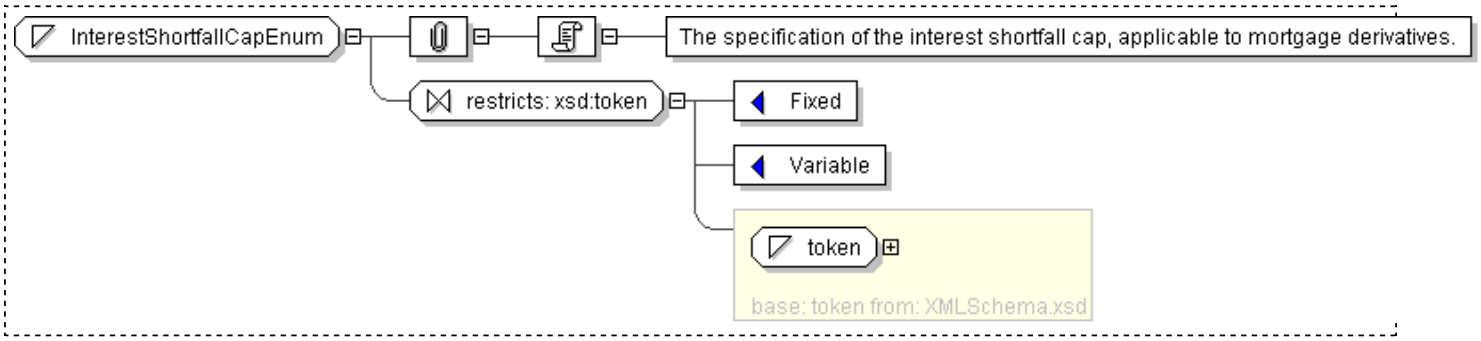
Simple Type: InterestShortfallCapEnum

[Table of contents]

Super-types:	xsd:token < InterestShortfallCapEnum (by restriction)
Sub-types:	None

Name	InterestShortfallCapEnum
Content	• Base XSD Type: token • *value* comes from list: {'Fixed'|'Variable'}
Documentation	The specification of the interest shortfall cap, applicable to mortgage derivatives.

Diagram



Schema Component Representation

```
<xsd:simpleType name="InterestShortfallCapEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Fixed"/>
    <xsd:enumeration value="Variable"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

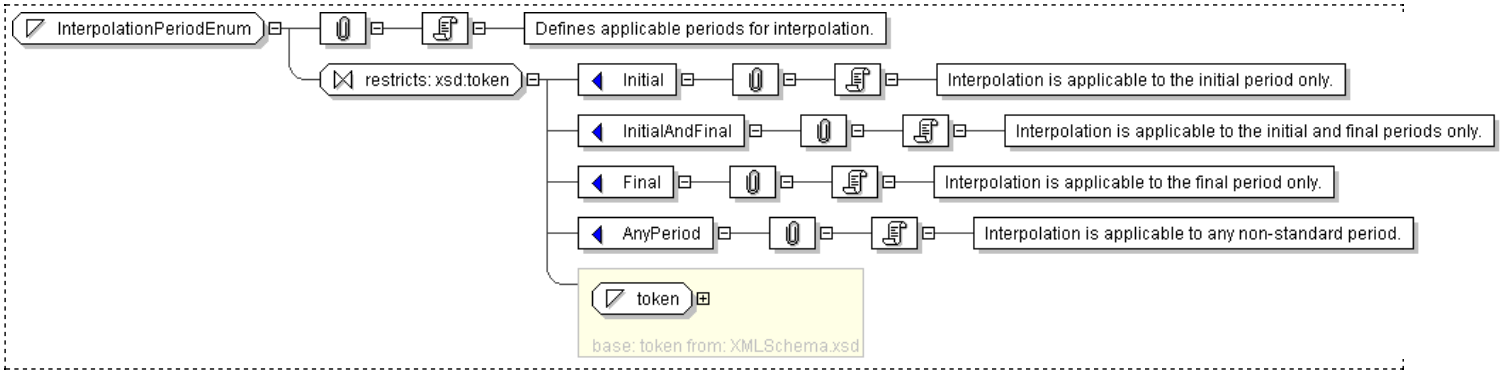
Simple Type: InterpolationPeriodEnum

[Table of contents]

Super-types:	xsd:token < InterpolationPeriodEnum (by restriction)
Sub-types:	None

Name	InterpolationPeriodEnum
Content	• Base XSD Type: token • *value* comes from list: {'Initial'|'InitialAndFinal'|'Final'|'AnyPeriod'}
Documentation	Defines applicable periods for interpolation.

Diagram



Schema Component Representation

```
<xsd:simpleType name="InterpolationPeriodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Initial"/>
    <xsd:enumeration value="InitialAndFinal"/>
    <xsd:enumeration value="Final"/>
    <xsd:enumeration value="AnyPeriod"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

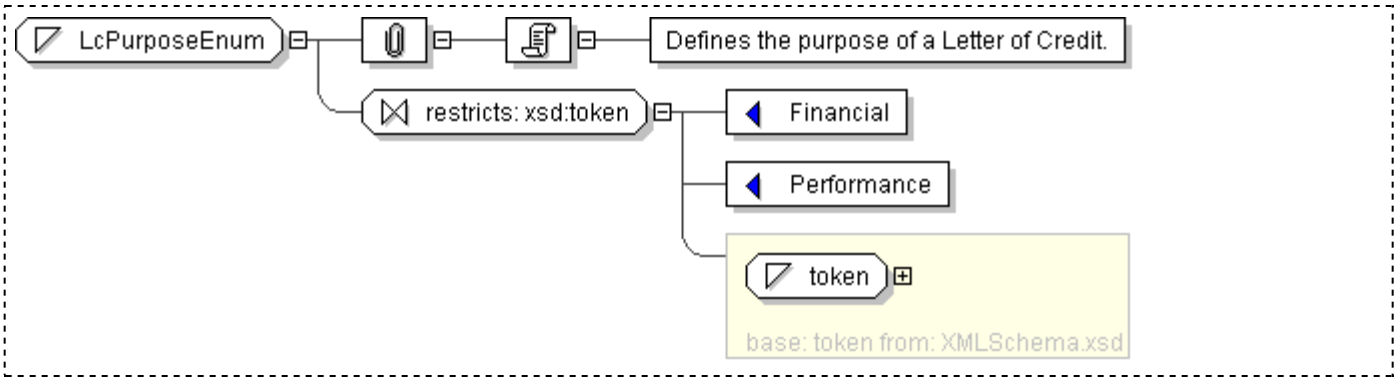
Simple Type: LcPurposeEnum

[Table of contents]

Super-types:	xsd:token < LcPurposeEnum (by restriction)
Sub-types:	None

Name	LcPurposeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Financial'|'Performance'}
Documentation	Defines the purpose of a Letter of Credit.

Diagram



Schema Component Representation

```
<xsd:simpleType name="LcPurposeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Financial"/>
    <xsd:enumeration value="Performance"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

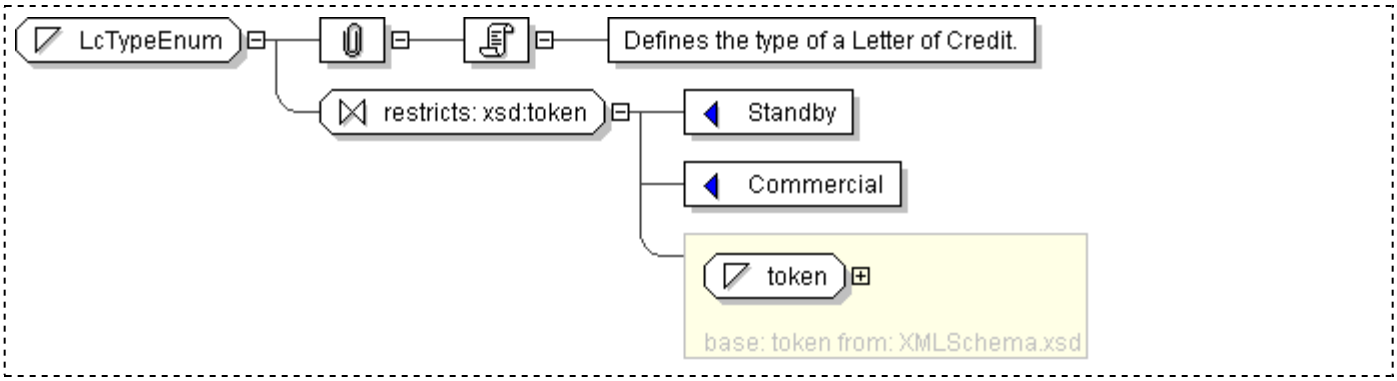
Simple Type: LcTypeEnum

[Table of contents]

Super-types:	xsd:token < LcTypeEnum (by restriction)
Sub-types:	None

Name	LcTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Standby'|'Commercial'}
Documentation	Defines the type of a Letter of Credit.

Diagram



Schema Component Representation

```
<xsd:simpleType name="LcTypeEnum">
   <xsd:restriction base=" xsd:token ">
      <xsd:enumeration value="Standby"/>
      <xsd:enumeration value="Commercial"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

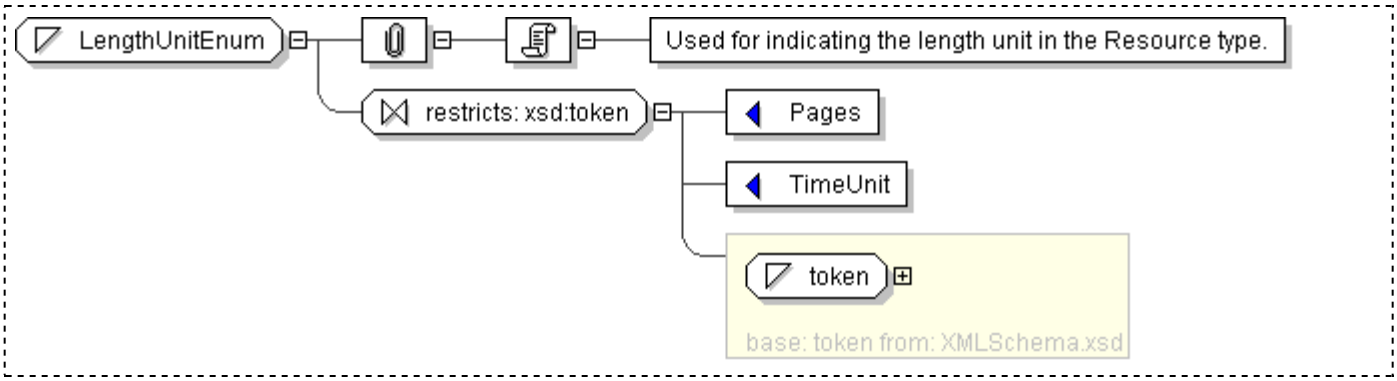
Simple Type: LengthUnitEnum

[Table of contents]

Super-types:	xsd:token < LengthUnitEnum (by restriction)
Sub-types:	None

Name	LengthUnitEnum
Content	• Base XSD Type: token • *value* comes from list: {'Pages'|'TimeUnit'}
Documentation	Used for indicating the length unit in the Resource type.

Diagram



Schema Component Representation

```
<xsd:simpleType name="LengthUnitEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Pages"/>
    <xsd:enumeration value="TimeUnit"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

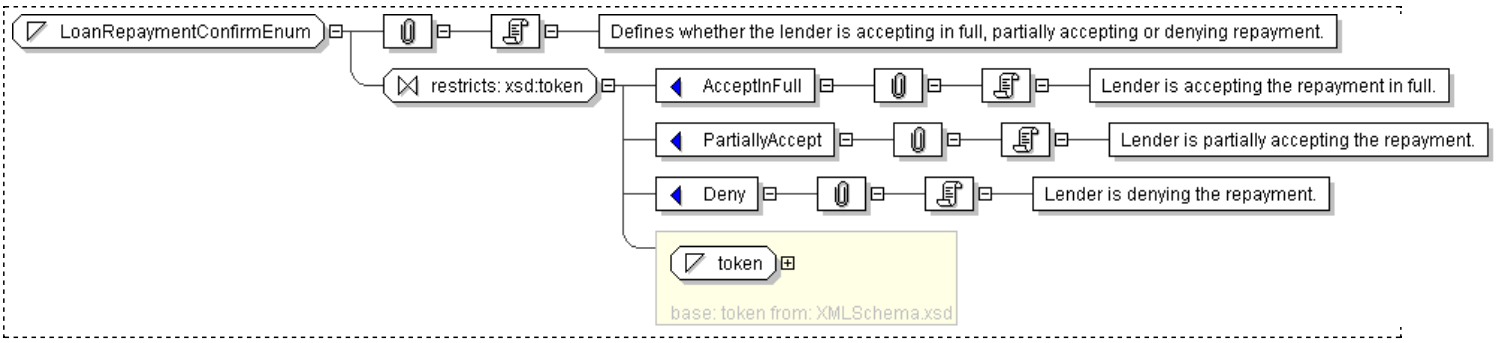
Simple Type: LoanRepaymentConfirmEnum

[Table of contents]

Super-types:	xsd:token < LoanRepaymentConfirmEnum (by restriction)
Sub-types:	None

Name	LoanRepaymentConfirmEnum
Content	• Base XSD Type: token • *value* comes from list: {'AcceptInFull'|'PartiallyAccept'|'Deny'}
Documentation	Defines whether the lender is accepting in full, partially accepting or denying repayment.

Diagram



Schema Component Representation

```
<xsd:simpleType name="LoanRepaymentConfirmEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AcceptInFull"/>
    <xsd:enumeration value="PartiallyAccept"/>
    <xsd:enumeration value="Deny"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

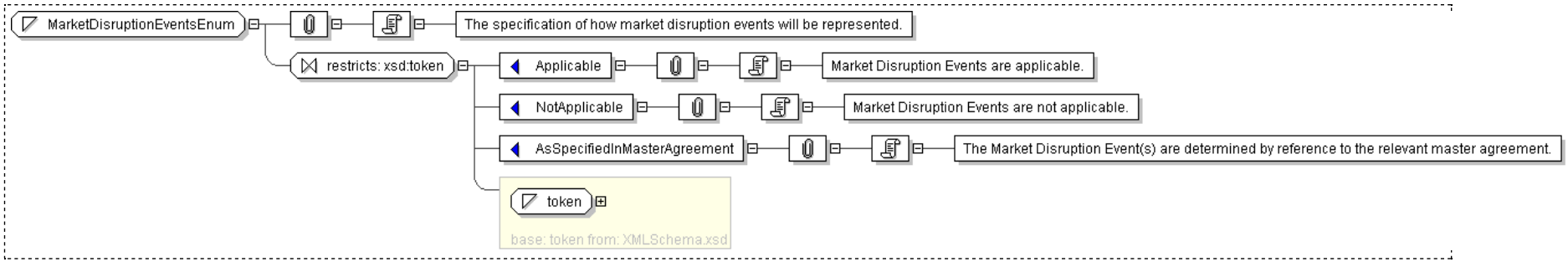
Simple Type: MarketDisruptionEventsEnum

[Table of contents]

Super-types:	xsd:token < MarketDisruptionEventsEnum (by restriction)
Sub-types:	None

Name	MarketDisruptionEventsEnum
Content	• Base XSD Type: token • *value* comes from list: {'Applicable'|'NotApplicable'|'AsSpecifiedInMasterAgreement'}
Documentation	The specification of how market disruption events will be represented.

Diagram



Schema Component Representation

```
<xsd:simpleType name="MarketDisruptionEventsEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Applicable"/>
    <xsd:enumeration value="NotApplicable"/>
    <xsd:enumeration value="AsSpecifiedInMasterAgreement"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

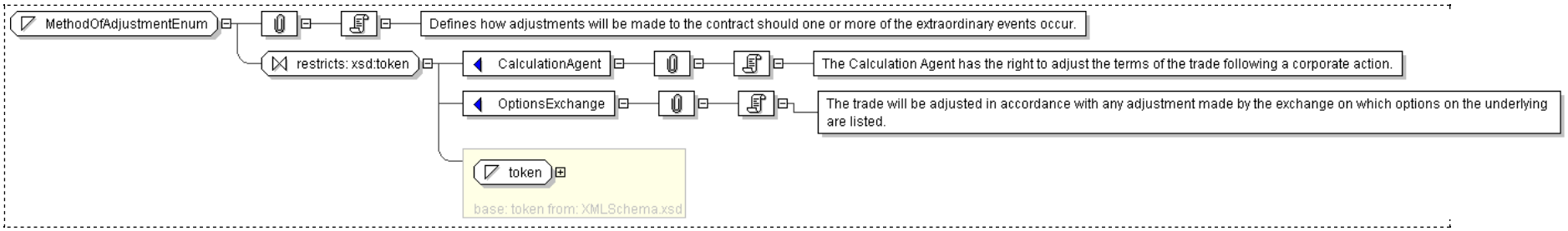
Simple Type: MethodOfAdjustmentEnum

[Table of contents]

Super-types:	xsd:token < MethodOfAdjustmentEnum (by restriction)
Sub-types:	None

Name	MethodOfAdjustmentEnum
Content	• Base XSD Type: token • *value* comes from list: {'CalculationAgent'|'OptionsExchange'}
Documentation	Defines how adjustments will be made to the contract should one or more of the extraordinary events occur.

Diagram



Schema Component Representation

```
<xsd:simpleType name="MethodOfAdjustmentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CalculationAgent"/>
    <xsd:enumeration value="OptionsExchange"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

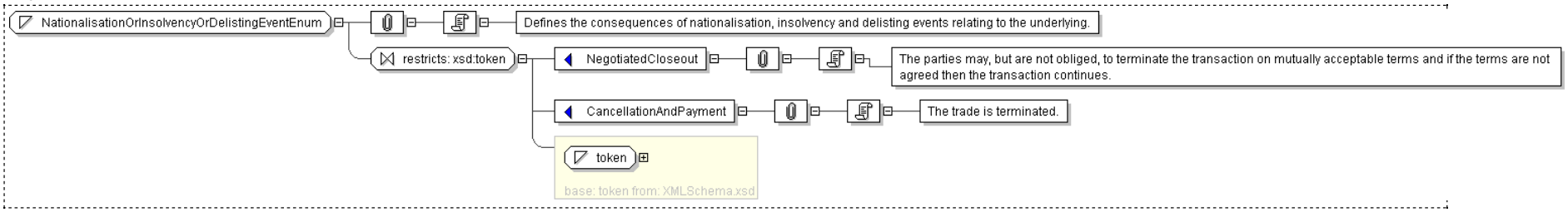
Simple Type: NationalisationOrInsolvencyOrDelistingEventEnum

[Table of contents]

Super-types:	xsd:token < NationalisationOrInsolvencyOrDelistingEventEnum (by restriction)
Sub-types:	None

Name	NationalisationOrInsolvencyOrDelistingEventEnum
Content	• Base XSD Type: token • *value* comes from list: {'NegotiatedCloseout'|'CancellationAndPayment'}
Documentation	Defines the consequences of nationalisation, insolvency and delisting events relating to the underlying.

Diagram



Schema Component Representation

```
<xsd:simpleType name="NationalisationOrInsolvencyOrDelistingEventEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="NegotiatedCloseout"/>
    <xsd:enumeration value="CancellationAndPayment"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

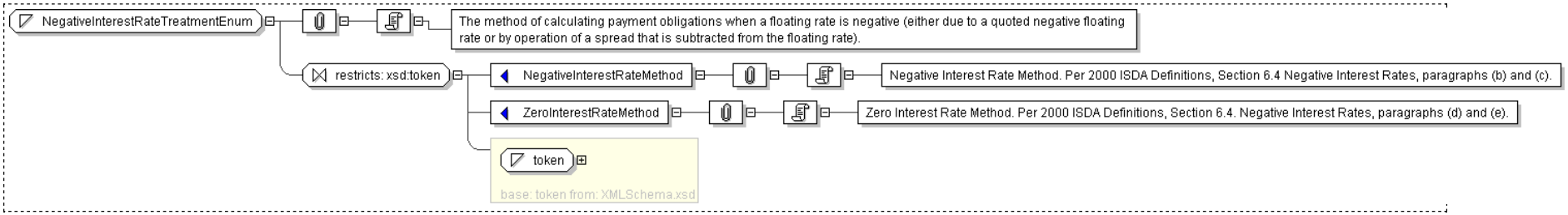
Simple Type: NegativeInterestRateTreatmentEnum

[Table of contents]

Super-types:	xsd:token < NegativeInterestRateTreatmentEnum (by restriction)
Sub-types:	None

Name	NegativeInterestRateTreatmentEnum
Content	• Base XSD Type: token • *value* comes from list: {'NegativeInterestRateMethod'|'ZeroInterestRateMethod'}
Documentation	The method of calculating payment obligations when a floating rate is negative (either due to a quoted negative floating rate or by operation of a spread that is subtracted from the floating rate).

Diagram



Schema Component Representation

```
<xsd:simpleType name="NegativeInterestRateTreatmentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="NegativeInterestRateMethod"/>
    <xsd:enumeration value="ZeroInterestRateMethod"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

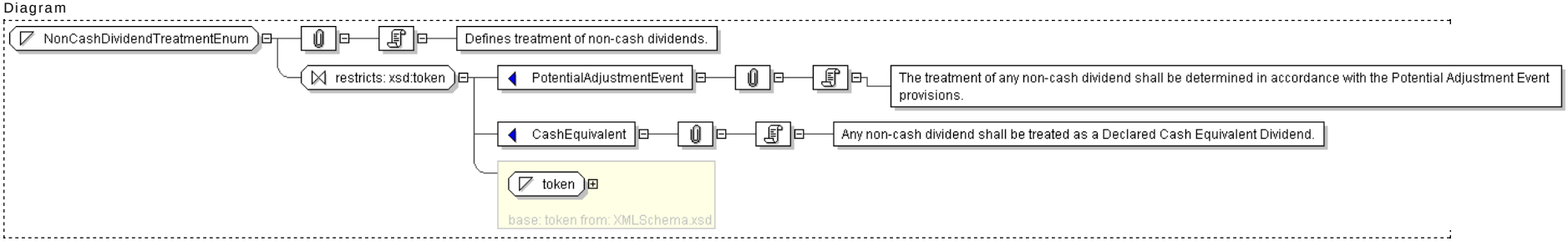
Simple Type: NonCashDividendTreatmentEnum

[Table of contents]

Super-types:	xsd:token < NonCashDividendTreatmentEnum (by restriction)
Sub-types:	None

Name	NonCashDividendTreatmentEnum
Content	• Base XSD Type: token • *value* comes from list: {'PotentialAdjustmentEvent'|'CashEquivalent'}
Documentation	Defines treatment of non-cash dividends.

Diagram



Schema Component Representation

```
<xsd:simpleType name="NonCashDividendTreatmentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="PotentialAdjustmentEvent"/>
    <xsd:enumeration value="CashEquivalent"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

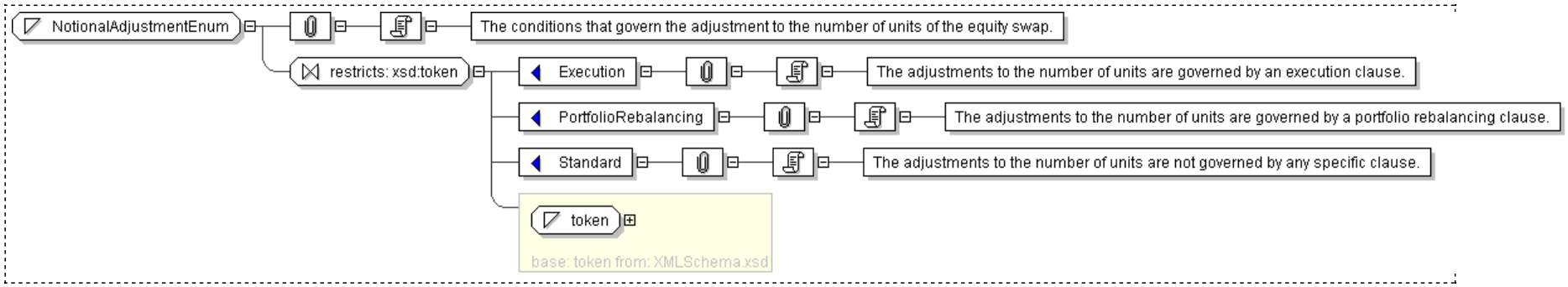
Simple Type: NotionalAdjustmentEnum

[Table of contents]

Super-types:	xsd:token < NotionalAdjustmentEnum (by restriction)
Sub-types:	None

Name	NotionalAdjustmentEnum
Content	• Base XSD Type: token • *value* comes from list: {'Execution'|'PortfolioRebalancing'|'Standard'}
Documentation	The conditions that govern the adjustment to the number of units of the equity swap.

Diagram



Schema Component Representation

```
<xsd:simpleType name="NotionalAdjustmentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Execution"/>
    <xsd:enumeration value="PortfolioRebalancing"/>
    <xsd:enumeration value="Standard"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

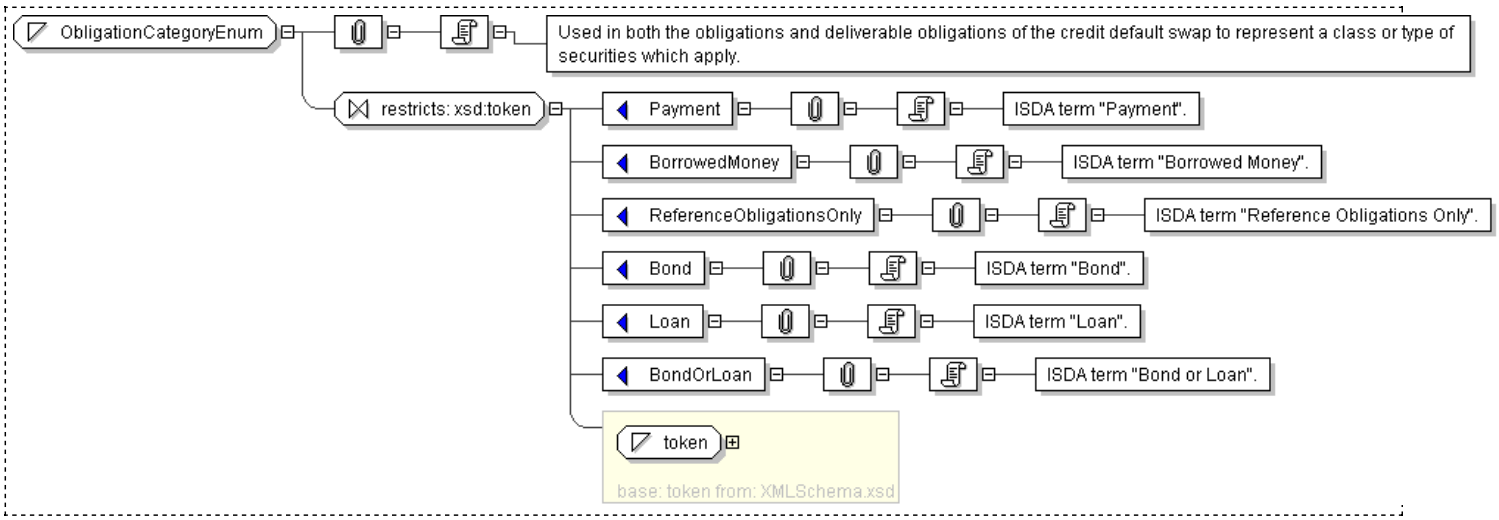
Simple Type: ObligationCategoryEnum

[Table of contents]

Super-types:	xsd:token < ObligationCategoryEnum (by restriction)
Sub-types:	None

Name	ObligationCategoryEnum
Content	• Base XSD Type: token • *value* comes from list: {'Payment'|'BorrowedMoney'|'ReferenceObligationsOnly'|'Bond'|'Loan'|'BondOrLoan'}
Documentation	Used in both the obligations and deliverable obligations of the credit default swap to represent a class or type of securities which apply.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ObligationCategoryEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Payment"/>
    <xsd:enumeration value="BorrowedMoney"/>
    <xsd:enumeration value="ReferenceObligationsOnly"/>
    <xsd:enumeration value="Bond"/>
    <xsd:enumeration value="Loan"/>
    <xsd:enumeration value="BondOrLoan"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

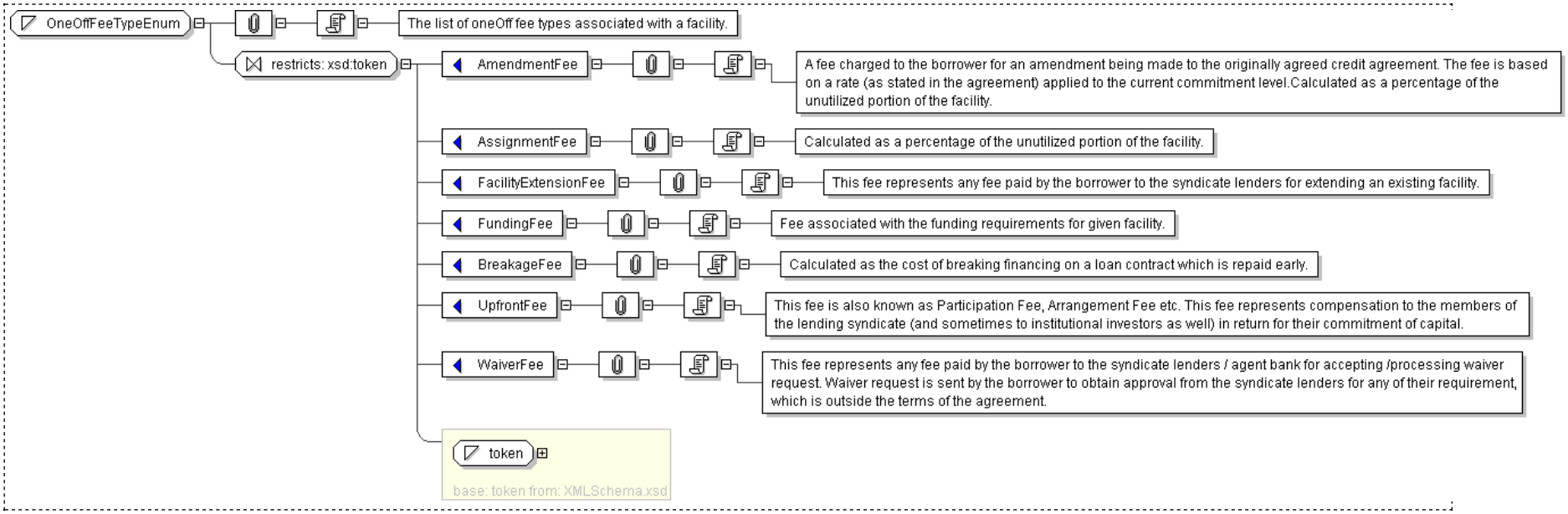
Simple Type: OneOffFeeTypeEnum

[Table of contents]

Super-types:	xsd:token < OneOffFeeTypeEnum (by restriction)
Sub-types:	None

Name	OneOffFeeTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'AmendmentFee'|'AssignmentFee'|'FacilityExtensionFee'|'FundingFee'|'BreakageFee'|'UpfrontFee'|'WaiverFee'}
Documentation	The list of oneOff fee types associated with a facility.

Diagram



Schema Component Representation

```
<xsd:simpleType name="OneOffFeeTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AmendmentFee"/>
    <xsd:enumeration value="AssignmentFee"/>
    <xsd:enumeration value="FacilityExtensionFee"/>
    <xsd:enumeration value="FundingFee"/>
    <xsd:enumeration value="BreakageFee"/>
    <xsd:enumeration value="UpfrontFee"/>
    <xsd:enumeration value="WaiverFee"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

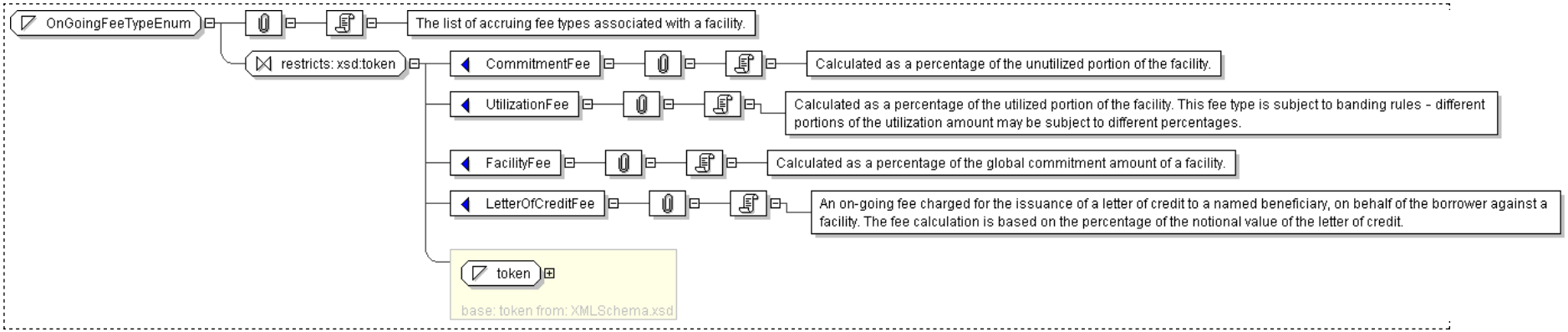
Simple Type: OnGoingFeeTypeEnum

[Table of contents]

Super-types:	xsd:token < OnGoingFeeTypeEnum (by restriction)
Sub-types:	None

Name	OnGoingFeeTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'CommitmentFee'|'UtilizationFee'|'FacilityFee'|'LetterOfCreditFee'}
Documentation	The list of accruing fee types associated with a facility.

Diagram



Schema Component Representation

```
<xsd:simpleType name="OnGoingFeeTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CommitmentFee"/>
    <xsd:enumeration value="UtilizationFee"/>
    <xsd:enumeration value="FacilityFee"/>
    <xsd:enumeration value="LetterOfCreditFee"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Simple Type: OptionTypeEnum

[Table of contents]

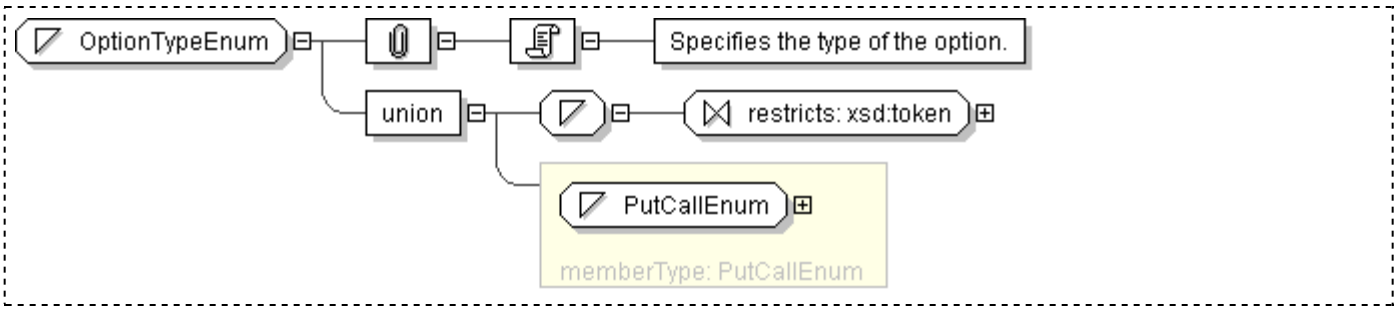
Super-types:	None
Sub-types:	None

Name	OptionTypeEnum

Content	• Union of following types:
	  ◦ PutCallEnum
	  ◦ Locally defined type:
	  ▪ Base XSD Type: token
	  ▪ *value* comes from list: {'Forward'|'Payer'|'Receiver'|'Straddle'}

Documentation	Specifies the type of the option.

Diagram



Schema Component Representation

```
<xsd:simpleType name="OptionTypeEnum">
  <xsd:union memberTypes=" PutCallEnum ">
    <xsd:simpleType>
      <xsd:restriction base=" xsd:token ">
        <xsd:enumeration value="Forward" deprecated="true" deprecatedReason="The
        optionType is to be used if the underlyer does not carry any mention of the
        resulting trade direction."/>
        <xsd:enumeration value="Payer"/>
        <xsd:enumeration value="Receiver"/>
        <xsd:enumeration value="Straddle"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

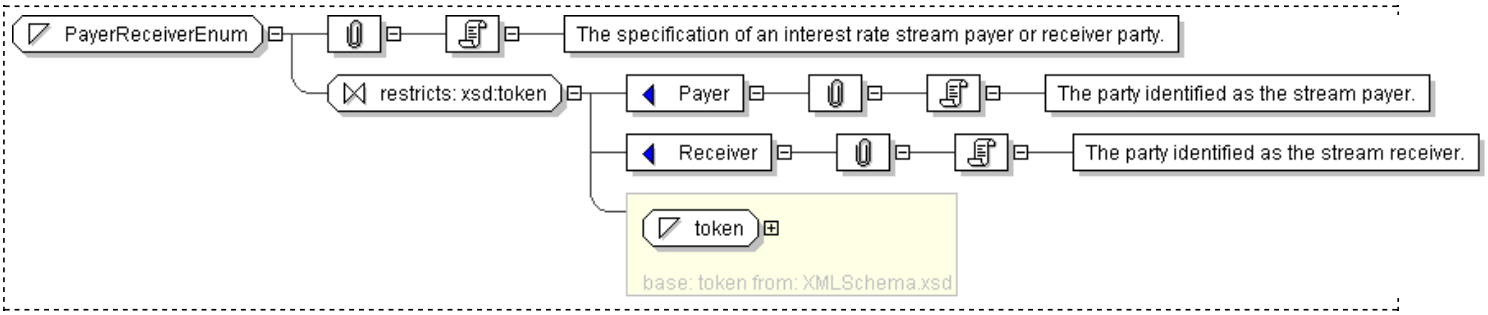
Simple Type: PayerReceiverEnum

[Table of contents]

Super-types:	xsd:token < PayerReceiverEnum (by restriction)
Sub-types:	None

Name	PayerReceiverEnum
Content	• Base XSD Type: token • *value* comes from list: {'Payer'|'Receiver'}
Documentation	The specification of an interest rate stream payer or receiver party.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PayerReceiverEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Payer"/>
    <xsd:enumeration value="Receiver"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

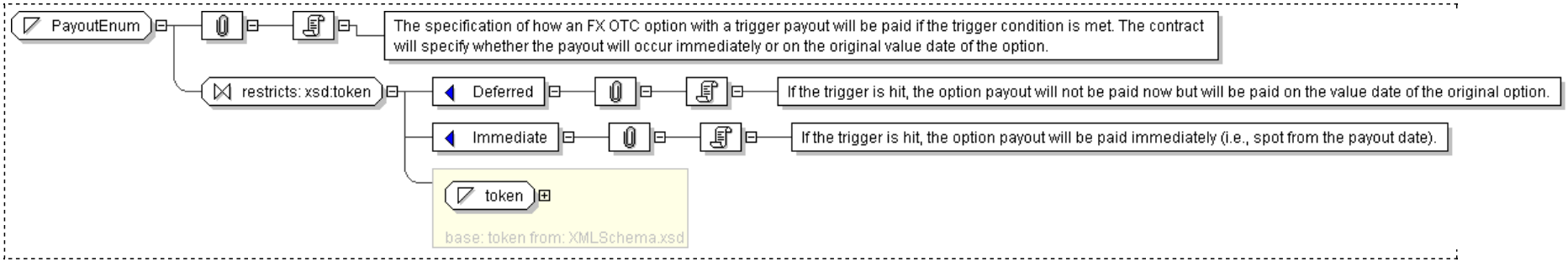
Simple Type: PayoutEnum

[Table of contents]

Super-types:	xsd:token < PayoutEnum (by restriction)
Sub-types:	None

Name	PayoutEnum
Content	• Base XSD Type: token • *value* comes from list: {'Deferred'|'Immediate'}
Documentation	The specification of how an FX OTC option with a trigger payout will be paid if the trigger condition is met. The contract will specify whether the payout will occur immediately or on the original value date of the option.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PayoutEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Deferred"/>
    <xsd:enumeration value="Immediate"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

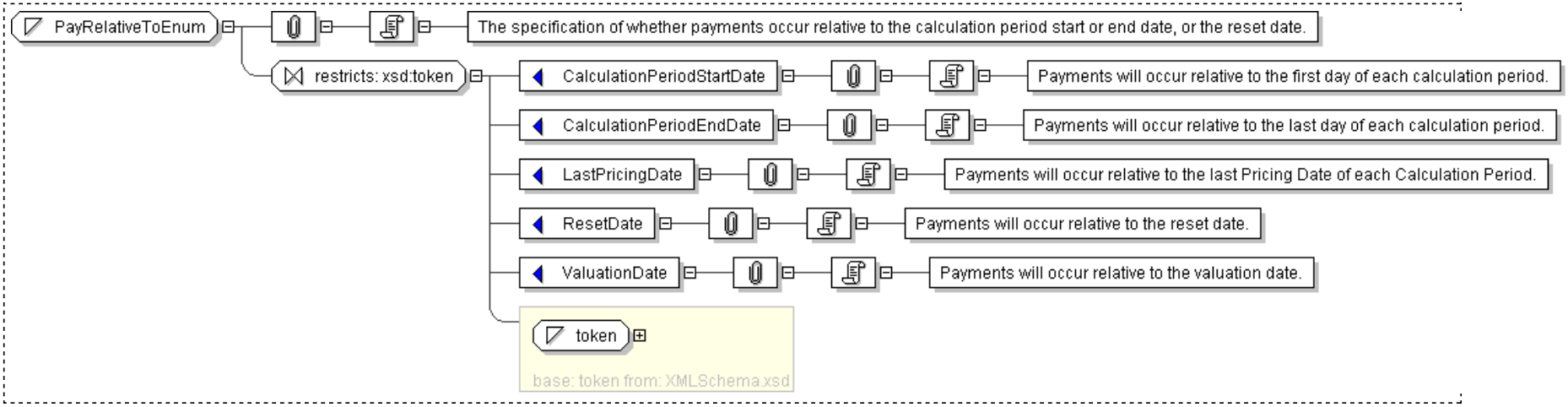
Simple Type: PayRelativeToEnum

[Table of contents]

Super-types:	xsd:token < PayRelativeToEnum (by restriction)
Sub-types:	None

Name	PayRelativeToEnum
Content	• Base XSD Type: token • *value* comes from list: {'CalculationPeriodStartDate'|'CalculationPeriodEndDate'|'LastPricingDate'|'ResetDate'|'ValuationDate'}
Documentation	The specification of whether payments occur relative to the calculation period start or end date, or the reset date.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PayRelativeToEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CalculationPeriodStartDate"/>
    <xsd:enumeration value="CalculationPeriodEndDate"/>
    <xsd:enumeration value="LastPricingDate"/>
    <xsd:enumeration value="ResetDate"/>
    <xsd:enumeration value="ValuationDate"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

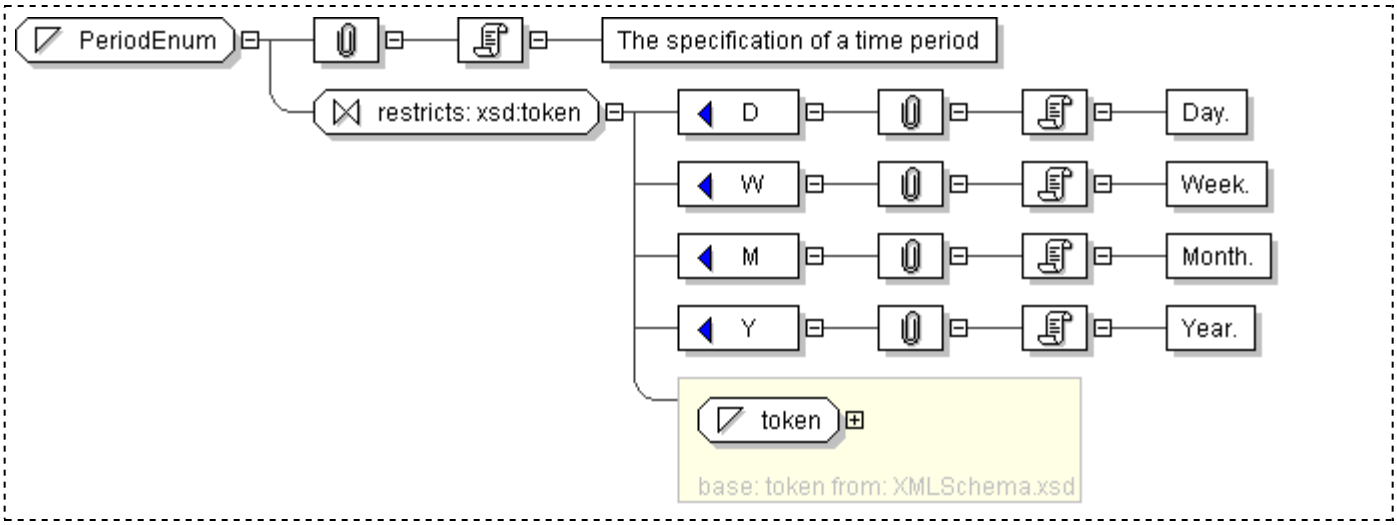
Simple Type: PeriodEnum

[Table of contents]

Super-types:	xsd:token < PeriodEnum (by restriction)
Sub-types:	None

Name	PeriodEnum
Content	• Base XSD Type: token • *value* comes from list: {'D'|'W'|'M'|'Y'}
Documentation	The specification of a time period

Diagram



Schema Component Representation

```
<xsd:simpleType name="PeriodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="D"/>
    <xsd:enumeration value="W"/>
    <xsd:enumeration value="M"/>
    <xsd:enumeration value="Y"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Simple Type: PeriodExtendedEnum

[Table of contents]

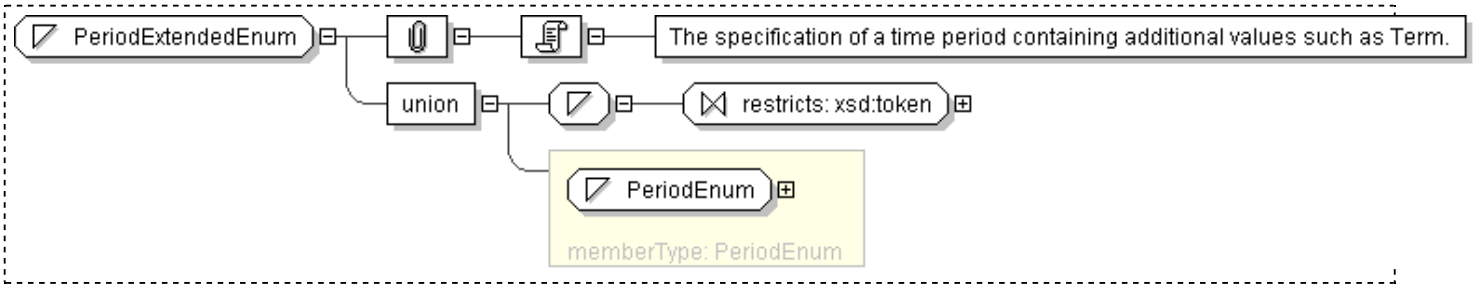
Super-types:	None
Sub-types:	None

Name	PeriodExtendedEnum

Content	• Union of following types:
	  ◦ PeriodEnum
	  ◦ Locally defined type:
	  ▪ Base XSD Type: token
	  ▪ *value* comes from list: {'T'}

Documentation	The specification of a time period containing additional values such as Term.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PeriodExtendedEnum">
  <xsd:union memberTypes=" PeriodEnum ">
    <xsd:simpleType>
      <xsd:restriction base=" xsd:token ">
        <xsd:enumeration value="T"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

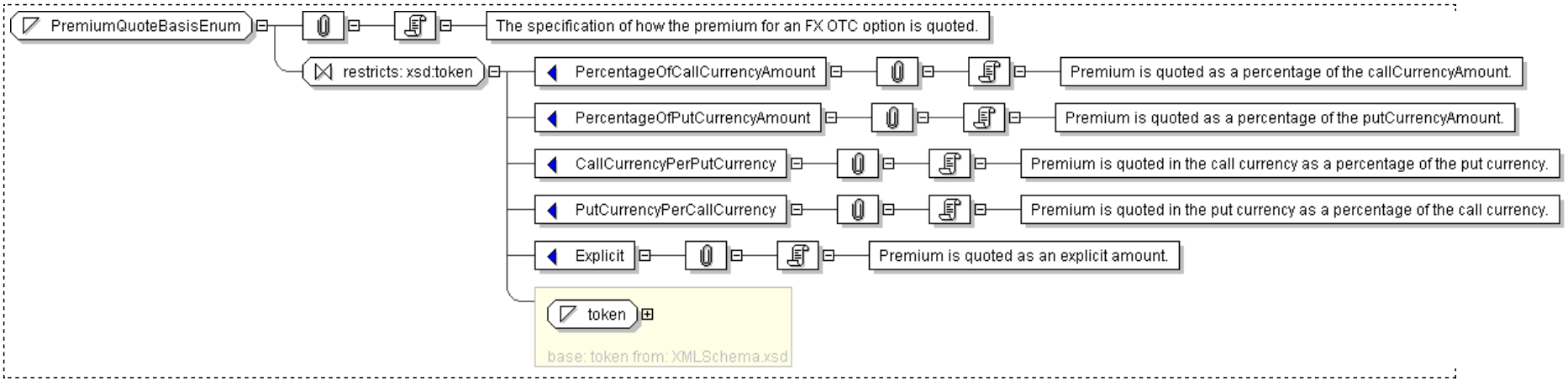
Simple Type: PremiumQuoteBasisEnum

[Table of contents]

Super-types:	xsd:token < PremiumQuoteBasisEnum (by restriction)
Sub-types:	None

Name	PremiumQuoteBasisEnum
Content	• Base XSD Type: token • *value* comes from list: {'PercentageOfCallCurrencyAmount'|'PercentageOfPutCurrencyAmount'|'CallCurrencyPerPutCurrency'|'PutCurrencyPerCallCurrency'|'Explicit'}
Documentation	The specification of how the premium for an FX OTC option is quoted.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PremiumQuoteBasisEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="PercentageOfCallCurrencyAmount"/>
    <xsd:enumeration value="PercentageOfPutCurrencyAmount"/>
    <xsd:enumeration value="CallCurrencyPerPutCurrency"/>
    <xsd:enumeration value="PutCurrencyPerCallCurrency"/>
    <xsd:enumeration value="Explicit"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

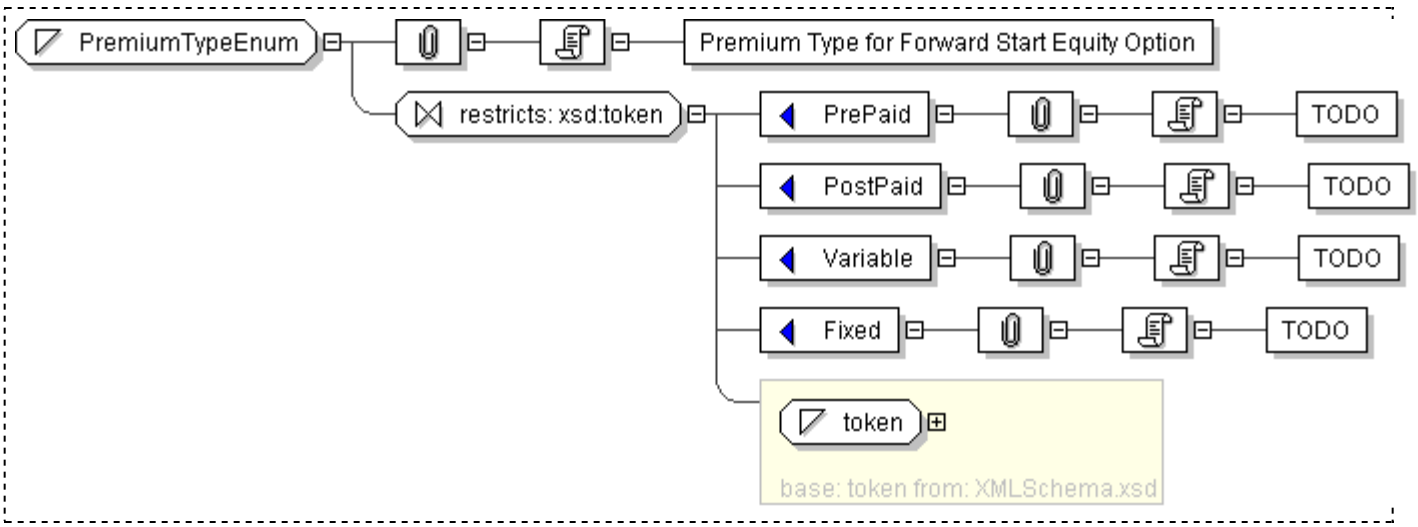
Simple Type: PremiumTypeEnum

[Table of contents]

Super-types:	xsd:token < PremiumTypeEnum (by restriction)
Sub-types:	None

Name	PremiumTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'PrePaid'|'PostPaid'|'Variable'|'Fixed'}
Documentation	Premium Type for Forward Start Equity Option

Diagram



Schema Component Representation

```
<xsd:simpleType name="PremiumTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="PrePaid"/>
    <xsd:enumeration value="PostPaid"/>
    <xsd:enumeration value="Variable"/>
    <xsd:enumeration value="Fixed"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

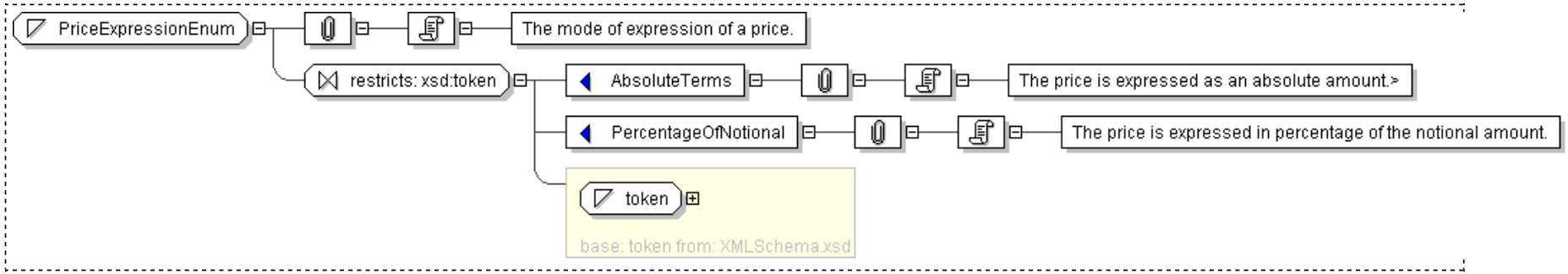
Simple Type: PriceExpressionEnum

[Table of contents]

Super-types:	xsd:token < PriceExpressionEnum (by restriction)
Sub-types:	None

Name	PriceExpressionEnum
Content	• Base XSD Type: token • *value* comes from list: {'AbsoluteTerms'|'PercentageOfNotional'}
Documentation	The mode of expression of a price.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PriceExpressionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AbsoluteTerms"/>
    <xsd:enumeration value="PercentageOfNotional"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

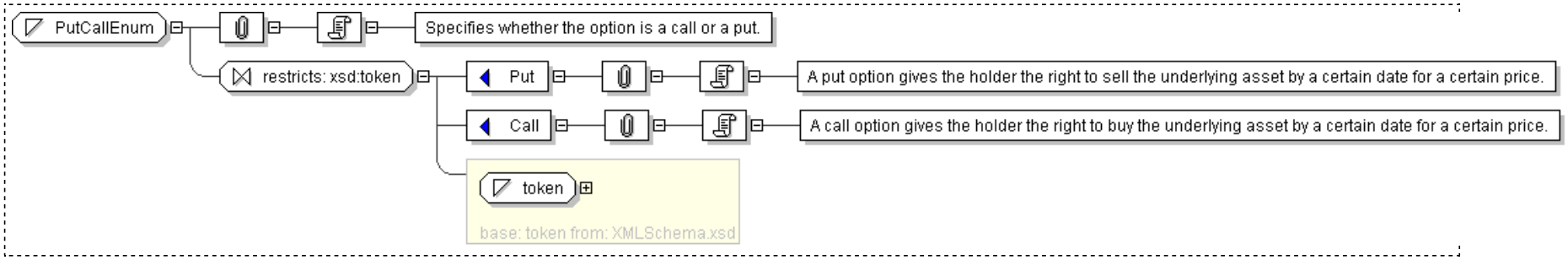
Simple Type: PutCallEnum

[Table of contents]

Super-types:	xsd:token < PutCallEnum (by restriction)
Sub-types:	None

Name	PutCallEnum
Content	• Base XSD Type: token • *value* comes from list: {'Put'|'Call'}
Documentation	Specifies whether the option is a call or a put.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PutCallEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Put"/>
    <xsd:enumeration value="Call"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

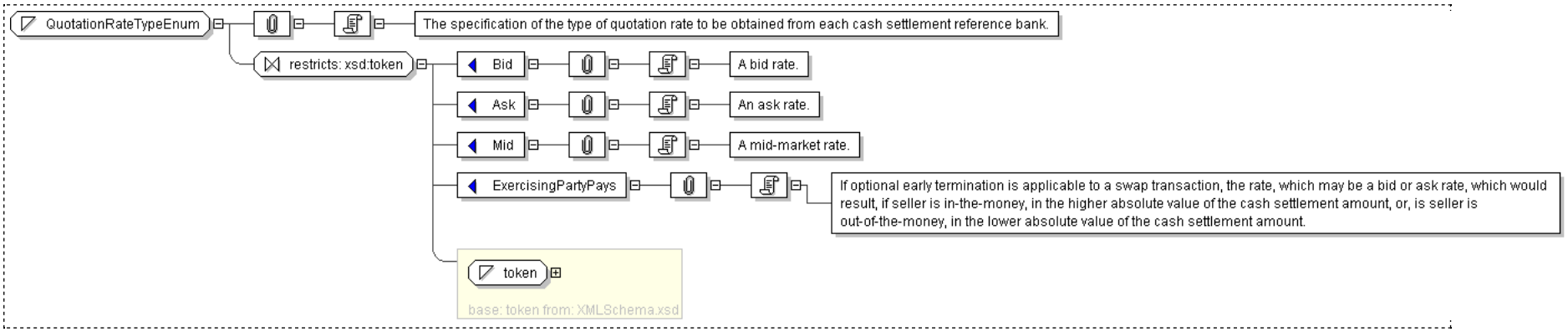
Simple Type: QuotationRateTypeEnum

[Table of contents]

Super-types:	xsd:token < QuotationRateTypeEnum (by restriction)
Sub-types:	None

Name	QuotationRateTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Bid'|'Ask'|'Mid'|'ExercisingPartyPays'}
Documentation	The specification of the type of quotation rate to be obtained from each cash settlement reference bank.

Diagram



Schema Component Representation

```
<xsd:simpleType name="QuotationRateTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Bid"/>
    <xsd:enumeration value="Ask"/>
    <xsd:enumeration value="Mid"/>
    <xsd:enumeration value="ExercisingPartyPays"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

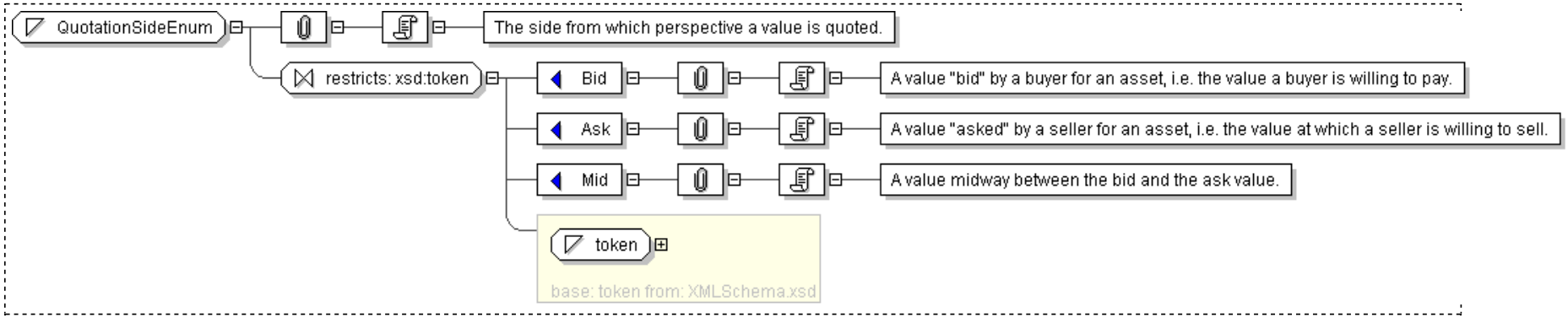
Simple Type: QuotationSideEnum

[Table of contents]

Super-types:	xsd:token < QuotationSideEnum (by restriction)
Sub-types:	None

Name	QuotationSideEnum
Content	• Base XSD Type: token • *value* comes from list: {'Bid'|'Ask'|'Mid'}
Documentation	The side from which perspective a value is quoted.

Diagram



Schema Component Representation

```
<xsd:simpleType name="QuotationSideEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Bid"/>
    <xsd:enumeration value="Ask"/>
    <xsd:enumeration value="Mid"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

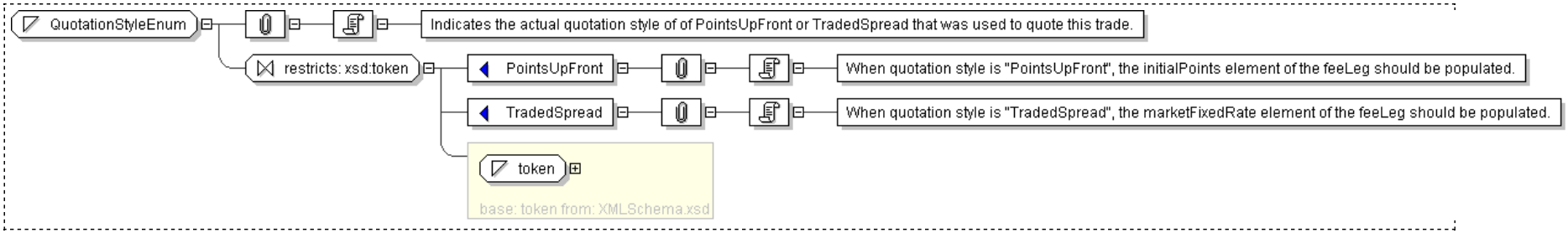
Simple Type: QuotationStyleEnum

[Table of contents]

Super-types:	xsd:token < QuotationStyleEnum (by restriction)
Sub-types:	None

Name	QuotationStyleEnum
Content	• Base XSD Type: token • *value* comes from list: {'PointsUpFront'|'TradedSpread'}
Documentation	Indicates the actual quotation style of of PointsUpFront or TradedSpread that was used to quote this trade.

Diagram



Schema Component Representation

```
<xsd:simpleType name="QuotationStyleEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="PointsUpFront"/>
    <xsd:enumeration value="TradedSpread"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

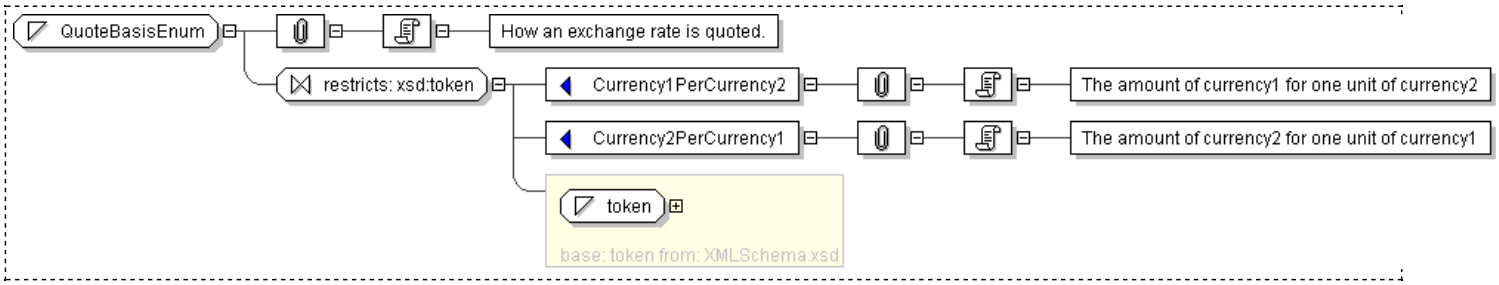
Simple Type: QuoteBasisEnum

[Table of contents]

Super-types:	xsd:token < QuoteBasisEnum (by restriction)
Sub-types:	None

Name	QuoteBasisEnum
Content	• Base XSD Type: token • *value* comes from list: {'Currency1PerCurrency2'|'Currency2PerCurrency1'}
Documentation	How an exchange rate is quoted.

Diagram



Schema Component Representation

```
<xsd:simpleType name="QuoteBasisEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Currency1PerCurrency2"/>
    <xsd:enumeration value="Currency2PerCurrency1"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

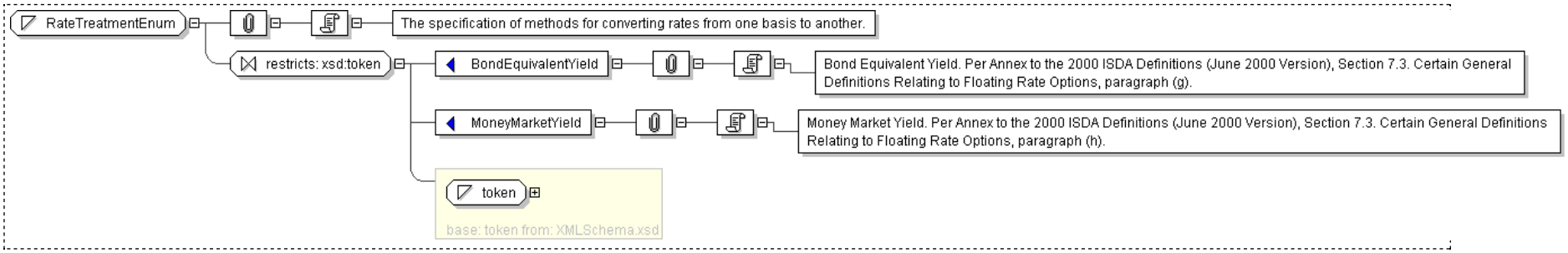
Simple Type: RateTreatmentEnum

[Table of contents]

Super-types:	xsd:token < RateTreatmentEnum (by restriction)
Sub-types:	None

Name	RateTreatmentEnum
Content	• Base XSD Type: token • *value* comes from list: {'BondEquivalentYield'|'MoneyMarketYield'}
Documentation	The specification of methods for converting rates from one basis to another.

Diagram



Schema Component Representation

```
<xsd:simpleType name="RateTreatmentEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="BondEquivalentYield"/>
    <xsd:enumeration value="MoneyMarketYield"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

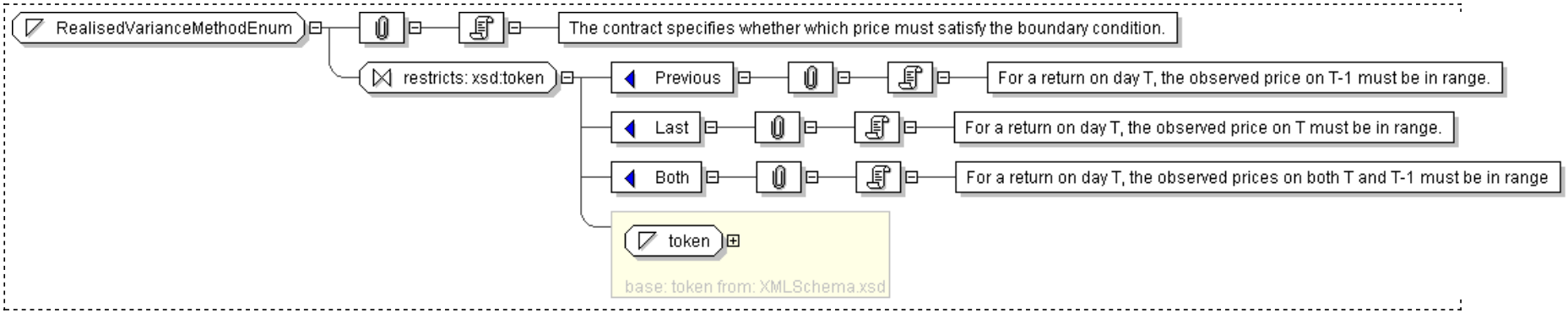
Simple Type: RealisedVarianceMethodEnum

[Table of contents]

Super-types:	xsd:token < RealisedVarianceMethodEnum (by restriction)
Sub-types:	None

Name	RealisedVarianceMethodEnum
Content	• Base XSD Type: token • *value* comes from list: {'Previous'|'Last'|'Both'}
Documentation	The contract specifies whether which price must satisfy the boundary condition.

Diagram



Schema Component Representation

```
<xsd:simpleType name="RealisedVarianceMethodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Previous"/>
    <xsd:enumeration value="Last"/>
    <xsd:enumeration value="Both"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

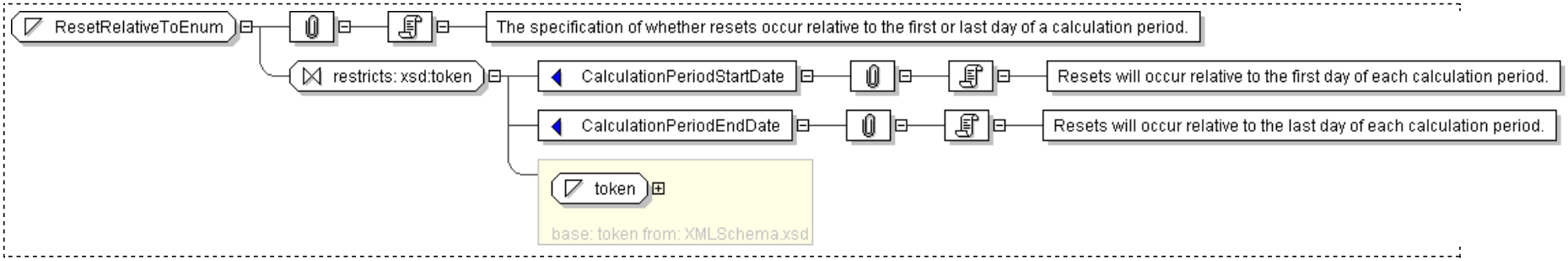
Simple Type: ResetRelativeToEnum

[Table of contents]

Super-types:	xsd:token < ResetRelativeToEnum (by restriction)
Sub-types:	None

Name	ResetRelativeToEnum
Content	• Base XSD Type: token • *value* comes from list: {'CalculationPeriodStartDate'|'CalculationPeriodEndDate'}
Documentation	The specification of whether resets occur relative to the first or last day of a calculation period.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ResetRelativeToEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="CalculationPeriodStartDate"/>
    <xsd:enumeration value="CalculationPeriodEndDate"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

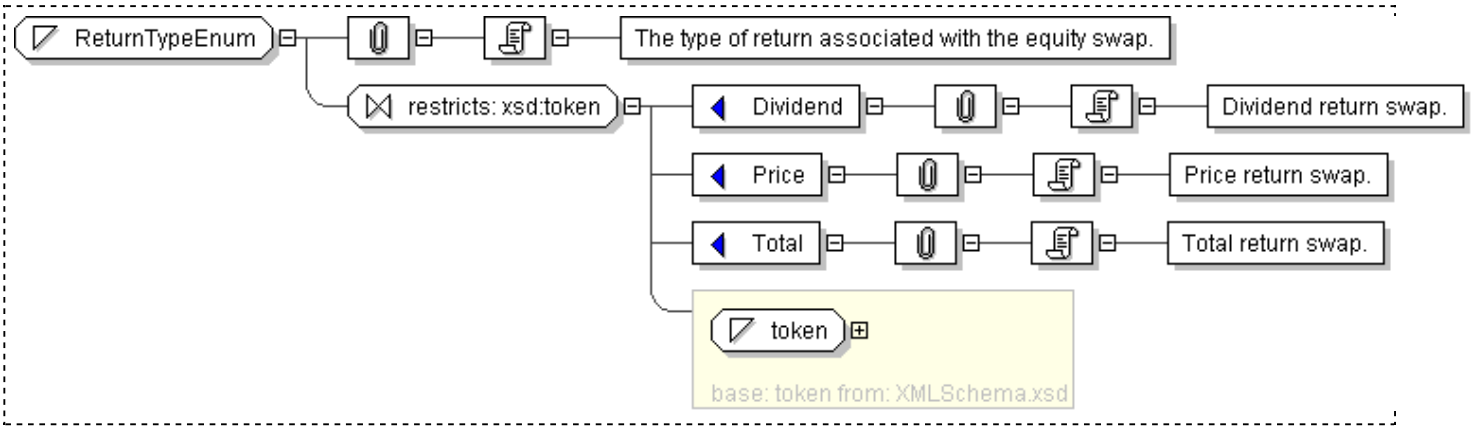
Simple Type: ReturnTypeEnum

[Table of contents]

Super-types:	xsd:token < ReturnTypeEnum (by restriction)
Sub-types:	None

Name	ReturnTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Dividend'|'Price'|'Total'}
Documentation	The type of return associated with the equity swap.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ReturnTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Dividend"/>
    <xsd:enumeration value="Price"/>
    <xsd:enumeration value="Total"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

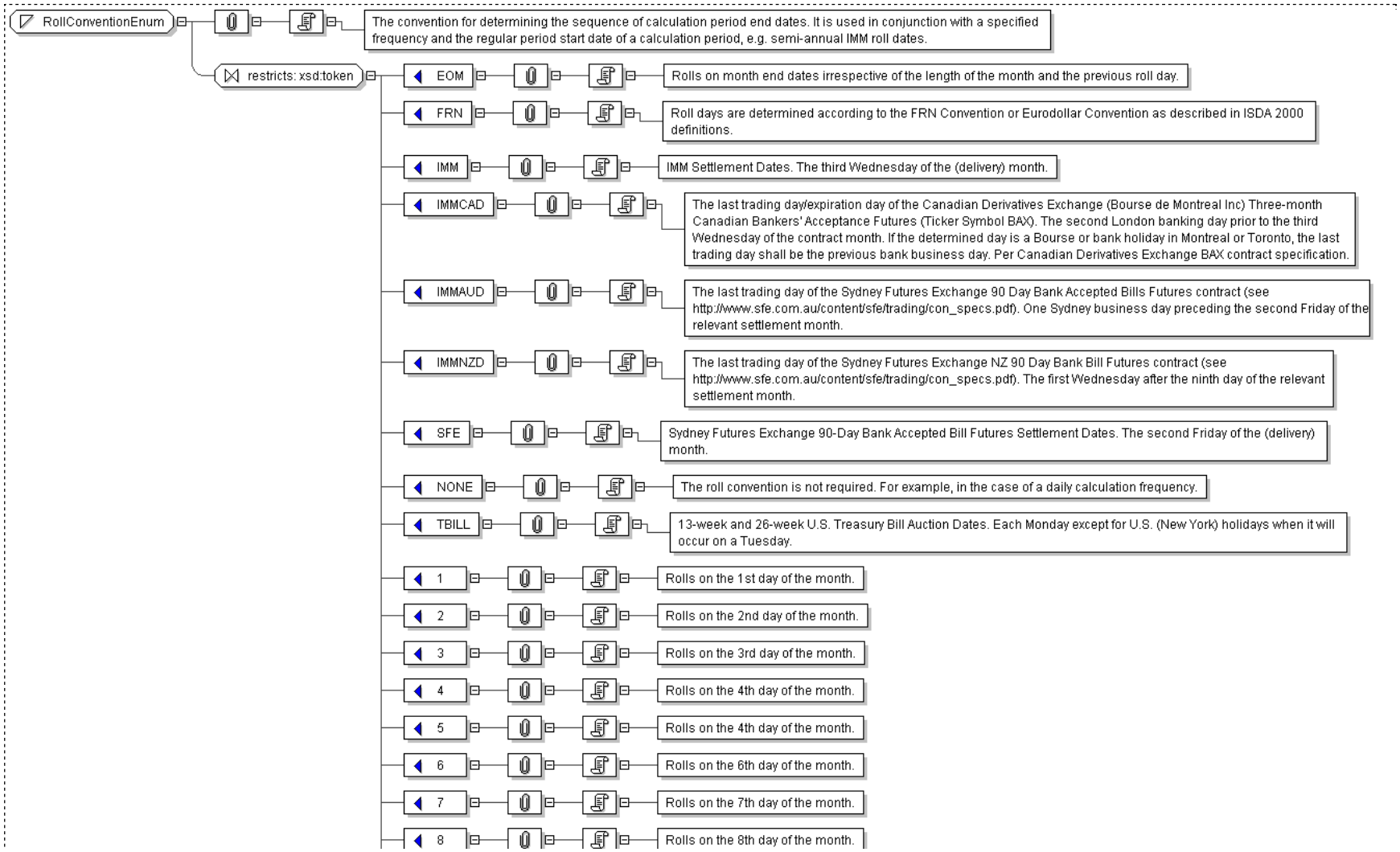
Simple Type: RollConventionEnum

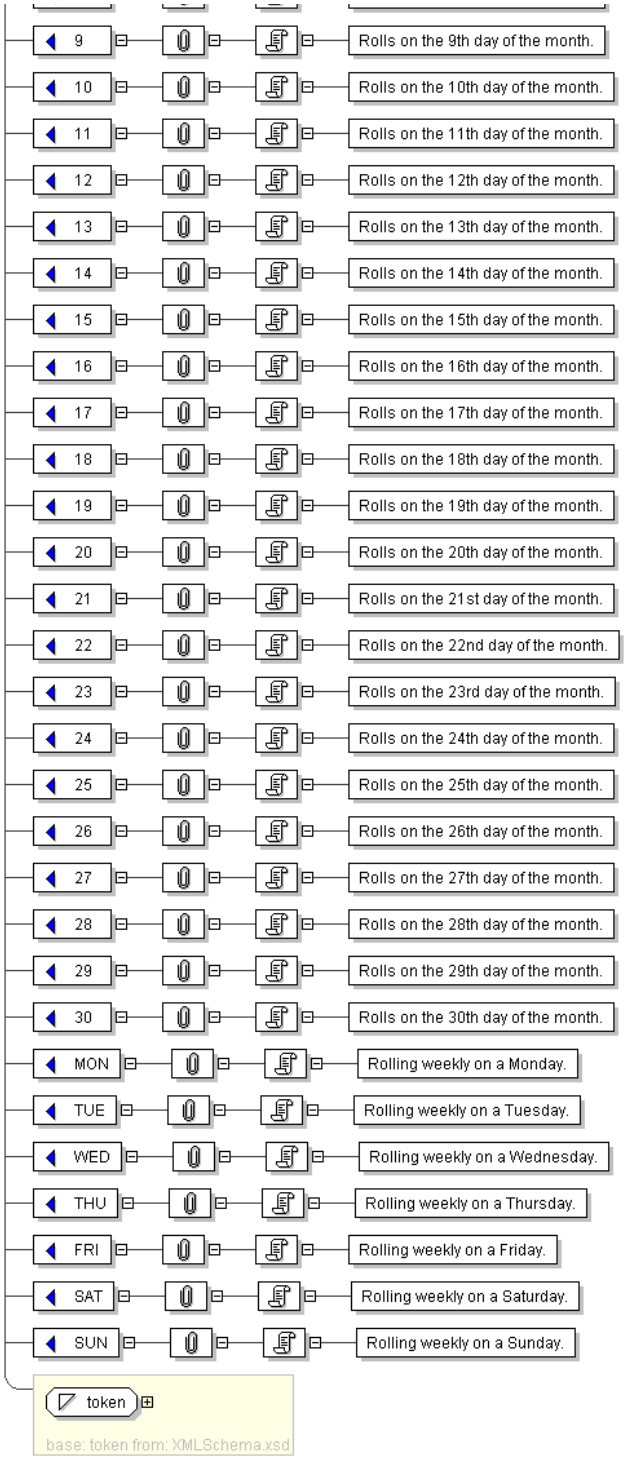
[Table of contents]

Super-types:	xsd:token < RollConventionEnum (by restriction)
Sub-types:	None

Name	RollConventionEnum
Content	• Base XSD Type: token • *value* comes from list: {'EOM'|'FRN'|'IMM'|'IMMCAD'|'IMMAUD'|'IMMNZD'|'SFE'|'NONE'|'TBILL'|'1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9'|'10'|'11'|'12'|'13'|'14'|'15'|'16'|'17'|'18'|'19'|'20'|'21'|'22'|'23'|'24'|'25'|'26'|'27'|'28'|'29'|'30'|'MON'|'TUE'|'WED'|'THU'|'FRI'|'SAT'|'SUN'}
Documentation	The convention for determining the sequence of calculation period end dates. It is used in conjunction with a specified frequency and the regular period start date of a calculation period, e.g. semi-annual IMM roll dates.

Diagram





Schema Component Representation

```
<xsd:simpleType name="RollConventionEnum">
```

```
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="EOM"/>
    <xsd:enumeration value="FRN"/>
    <xsd:enumeration value="IMM"/>
    <xsd:enumeration value="IMMCAD"/>
    <xsd:enumeration value="IMMAUD"/>
    <xsd:enumeration value="IMMNZD"/>
    <xsd:enumeration value="SFE"/>
    <xsd:enumeration value="NONE"/>
    <xsd:enumeration value="TBILL"/>
    <xsd:enumeration value="1"/>
    <xsd:enumeration value="2"/>
    <xsd:enumeration value="3"/>
    <xsd:enumeration value="4"/>
    <xsd:enumeration value="5"/>
    <xsd:enumeration value="6"/>
    <xsd:enumeration value="7"/>
    <xsd:enumeration value="8"/>
    <xsd:enumeration value="9"/>
    <xsd:enumeration value="10"/>
    <xsd:enumeration value="11"/>
    <xsd:enumeration value="12"/>
    <xsd:enumeration value="13"/>
    <xsd:enumeration value="14"/>
    <xsd:enumeration value="15"/>
    <xsd:enumeration value="16"/>
    <xsd:enumeration value="17"/>
    <xsd:enumeration value="18"/>
    <xsd:enumeration value="19"/>
    <xsd:enumeration value="20"/>
    <xsd:enumeration value="21"/>
    <xsd:enumeration value="22"/>
    <xsd:enumeration value="23"/>
    <xsd:enumeration value="24"/>
    <xsd:enumeration value="25"/>
    <xsd:enumeration value="26"/>
    <xsd:enumeration value="27"/>
    <xsd:enumeration value="28"/>
    <xsd:enumeration value="29"/>
    <xsd:enumeration value="30"/>
    <xsd:enumeration value="MON"/>
    <xsd:enumeration value="TUE"/>
    <xsd:enumeration value="WED"/>
    <xsd:enumeration value="THU"/>
    <xsd:enumeration value="FRI"/>
    <xsd:enumeration value="SAT"/>
    <xsd:enumeration value="SUN"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

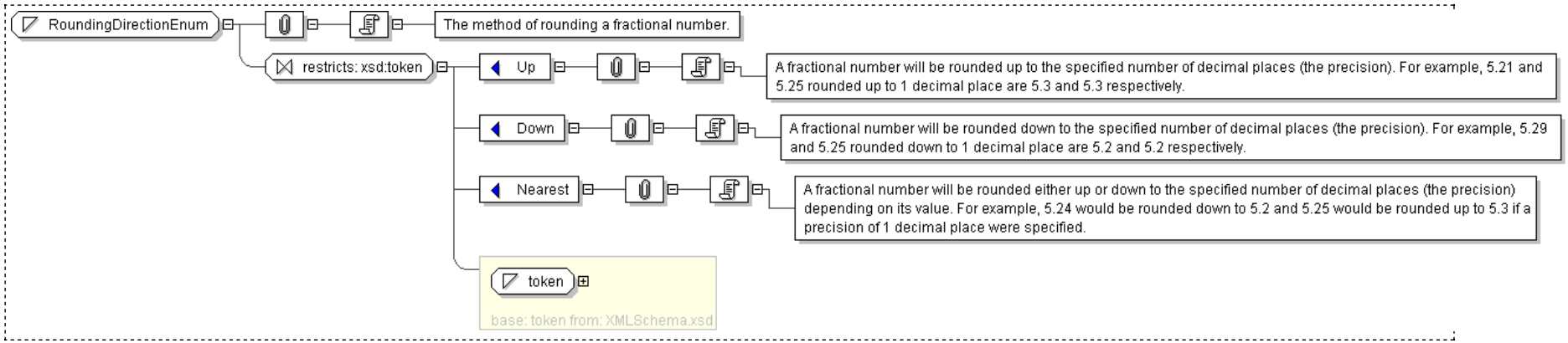
Simple Type: RoundingDirectionEnum

[Table of contents]

Super-types:	xsd:token < RoundingDirectionEnum (by restriction)
Sub-types:	None

Name	RoundingDirectionEnum
Content	• Base XSD Type: token • *value* comes from list: {'Up'|'Down'|'Nearest'}
Documentation	The method of rounding a fractional number.

Diagram



Schema Component Representation

```
<xsd:simpleType name="RoundingDirectionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Up"/>
    <xsd:enumeration value="Down"/>
    <xsd:enumeration value="Nearest"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

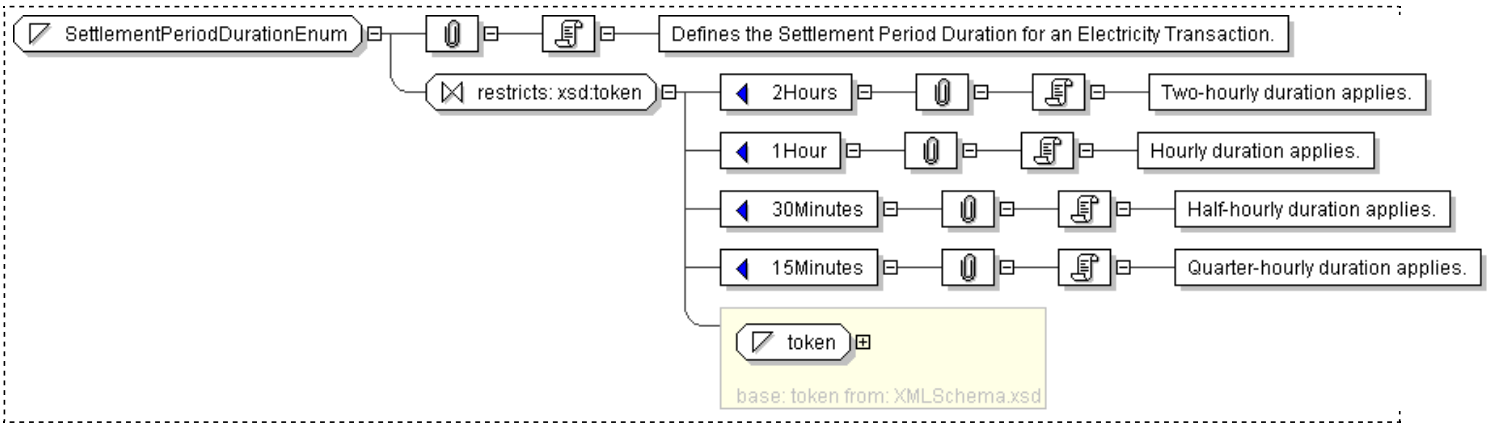
Simple Type: SettlementPeriodDurationEnum

[Table of contents]

Super-types:	xsd:token < SettlementPeriodDurationEnum (by restriction)
Sub-types:	None

Name	SettlementPeriodDurationEnum
Content	• Base XSD Type: token • *value* comes from list: {'2Hours'|'1Hour'|'30Minutes'|'15Minutes'}
Documentation	Defines the Settlement Period Duration for an Electricity Transaction.

Diagram



Schema Component Representation

```
<xsd:simpleType name="SettlementPeriodDurationEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="2Hours"/>
    <xsd:enumeration value="1Hour"/>
    <xsd:enumeration value="30Minutes"/>
    <xsd:enumeration value="15Minutes"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

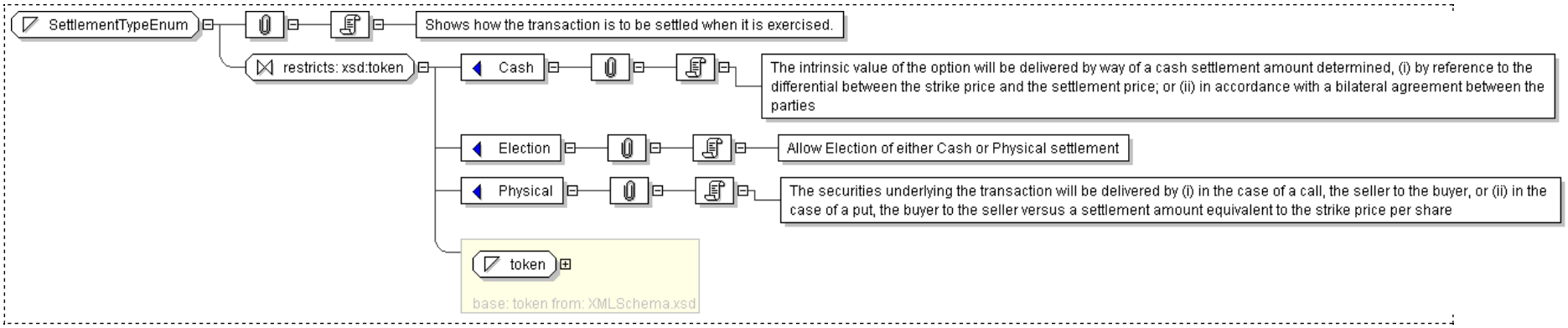
Simple Type: SettlementTypeEnum

[Table of contents]

Super-types:	xsd:token < SettlementTypeEnum (by restriction)
Sub-types:	None

Name	SettlementTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Cash'|'Election'|'Physical'}
Documentation	Shows how the transaction is to be settled when it is exercised.

Diagram



Schema Component Representation

```
<xsd:simpleType name="SettlementTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Cash"/>
    <xsd:enumeration value="Election"/>
    <xsd:enumeration value="Physical"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

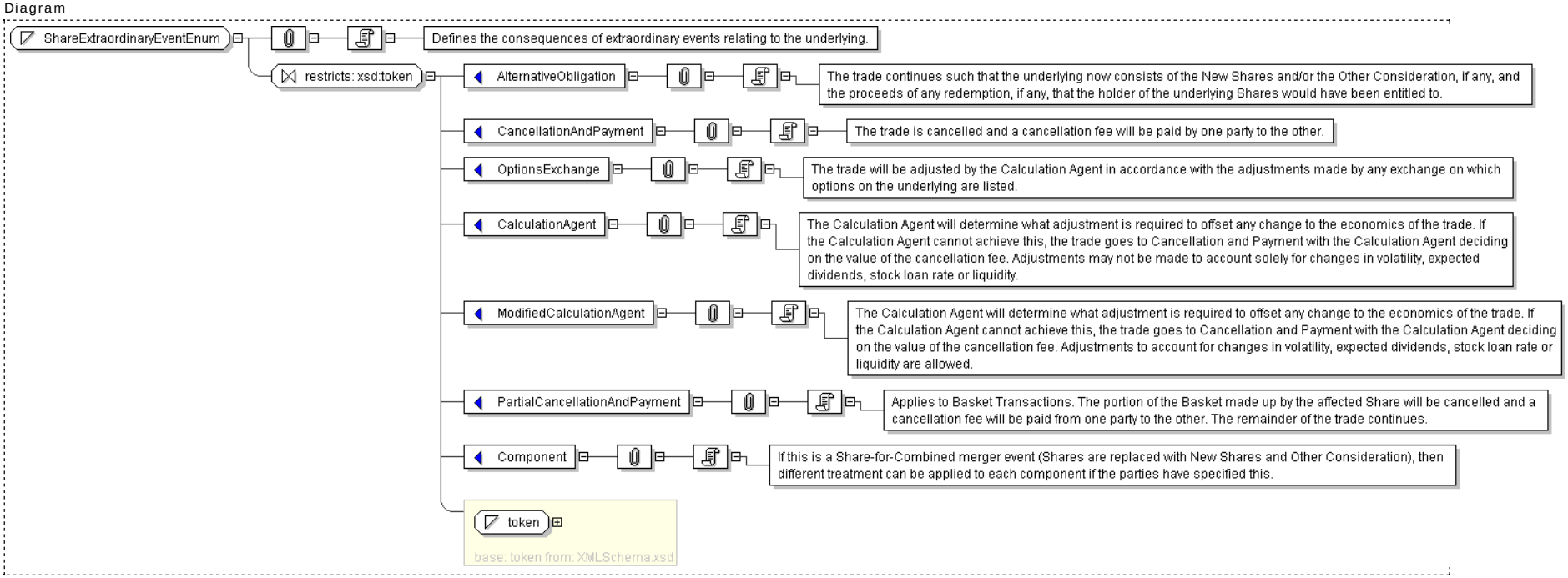
Simple Type: ShareExtraordinaryEventEnum

[Table of contents]

Super-types:	xsd:token < ShareExtraordinaryEventEnum (by restriction)
Sub-types:	None

Name	ShareExtraordinaryEventEnum
Content	• Base XSD Type: token • *value* comes from list: {'AlternativeObligation'|'CancellationAndPayment'|'OptionsExchange'|'CalculationAgent'|'ModifiedCalculationAgent'|'PartialCancellationAndPayment'|'Component'}
Documentation	Defines the consequences of extraordinary events relating to the underlying.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ShareExtraordinaryEventEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="AlternativeObligation"/>
    <xsd:enumeration value="CancellationAndPayment"/>
    <xsd:enumeration value="OptionsExchange"/>
    <xsd:enumeration value="CalculationAgent"/>
    <xsd:enumeration value="ModifiedCalculationAgent"/>
    <xsd:enumeration value="PartialCancellationAndPayment"/>
    <xsd:enumeration value="Component"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

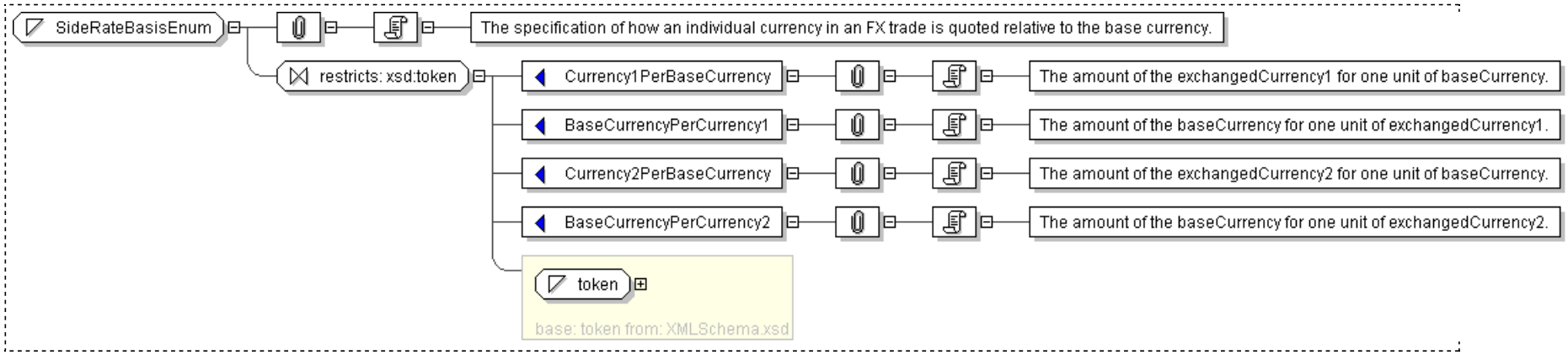
Simple Type: SideRateBasisEnum

[Table of contents]

Super-types:	xsd:token < SideRateBasisEnum (by restriction)
Sub-types:	None

Name	SideRateBasisEnum
Content	• Base XSD Type: token • *value* comes from list: {'Currency1PerBaseCurrency'|'BaseCurrencyPerCurrency1'|'Currency2PerBaseCurrency'|'BaseCurrencyPerCurrency2'}
Documentation	The specification of how an individual currency in an FX trade is quoted relative to the base currency.

Diagram



Schema Component Representation

```
<xsd:simpleType name="SideRateBasisEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Currency1PerBaseCurrency"/>
    <xsd:enumeration value="BaseCurrencyPerCurrency1"/>
    <xsd:enumeration value="Currency2PerBaseCurrency"/>
    <xsd:enumeration value="BaseCurrencyPerCurrency2"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

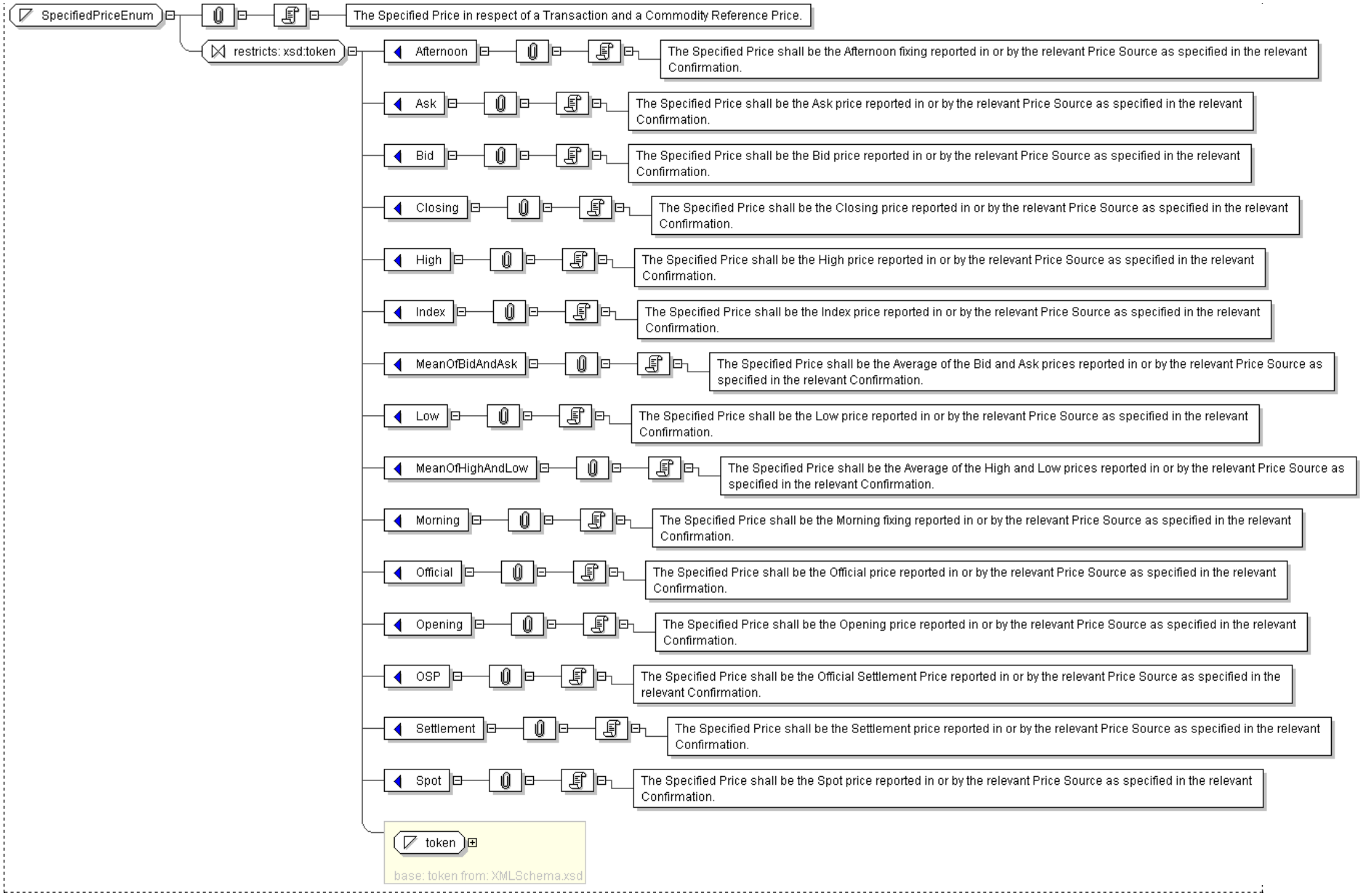
Simple Type: SpecifiedPriceEnum

[Table of contents]

Super-types:	xsd:token < SpecifiedPriceEnum (by restriction)
Sub-types:	None

Name	SpecifiedPriceEnum
Content	• Base XSD Type: token • *value* comes from list: {'Afternoon'|'Ask'|'Bid'|'Closing'|'High'|'Index'|'MeanOfBidAndAsk'|'Low'|'MeanOfHighAndLow'|'Morning'|'Official'|'Opening'|'OSP'|'Settlement'|'Spot'}
Documentation	The Specified Price in respect of a Transaction and a Commodity Reference Price.

Diagram



Schema Component Representation

```
<xsd:simpleType name="SpecifiedPriceEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Afternoon"/>
    <xsd:enumeration value="Ask"/>
    <xsd:enumeration value="Bid"/>
    <xsd:enumeration value="Closing"/>
    <xsd:enumeration value="High"/>
    <xsd:enumeration value="Index"/>
    <xsd:enumeration value="MeanOfBidAndAsk"/>
    <xsd:enumeration value="Low"/>
    <xsd:enumeration value="MeanOfHighAndLow"/>
    <xsd:enumeration value="Morning"/>
    <xsd:enumeration value="Official"/>
    <xsd:enumeration value="Opening"/>
```

```
      <xsd:enumeration value="OSP"/>
      <xsd:enumeration value="Settlement"/>
      <xsd:enumeration value="Spot"/>
    </xsd:restriction>
  </xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

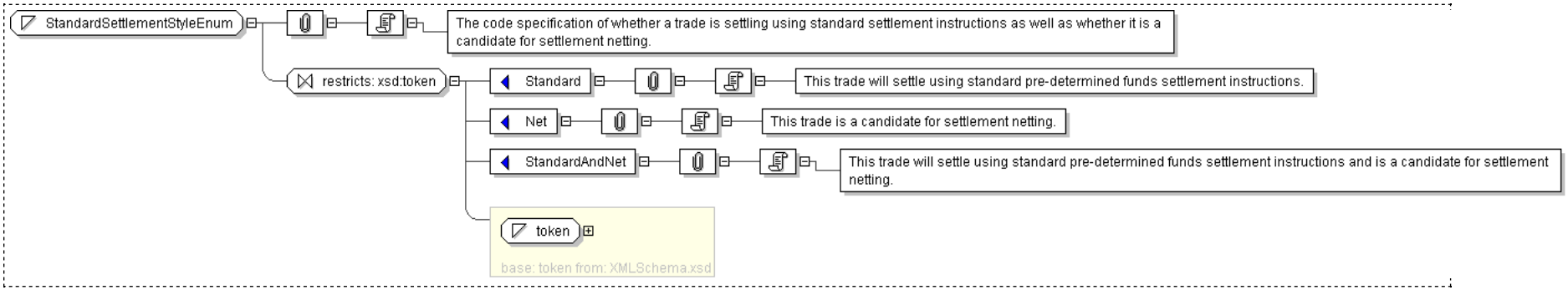
Simple Type: StandardSettlementStyleEnum

[Table of contents]

Super-types:	xsd:token < StandardSettlementStyleEnum (by restriction)
Sub-types:	None

Name	StandardSettlementStyleEnum
Content	• Base XSD Type: token • *value* comes from list: {'Standard'|'Net'|'StandardAndNet'}
Documentation	The code specification of whether a trade is settling using standard settlement instructions as well as whether it is a candidate for settlement netting.

Diagram



Schema Component Representation

```
<xsd:simpleType name="StandardSettlementStyleEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Standard"/>
    <xsd:enumeration value="Net"/>
    <xsd:enumeration value="StandardAndNet"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

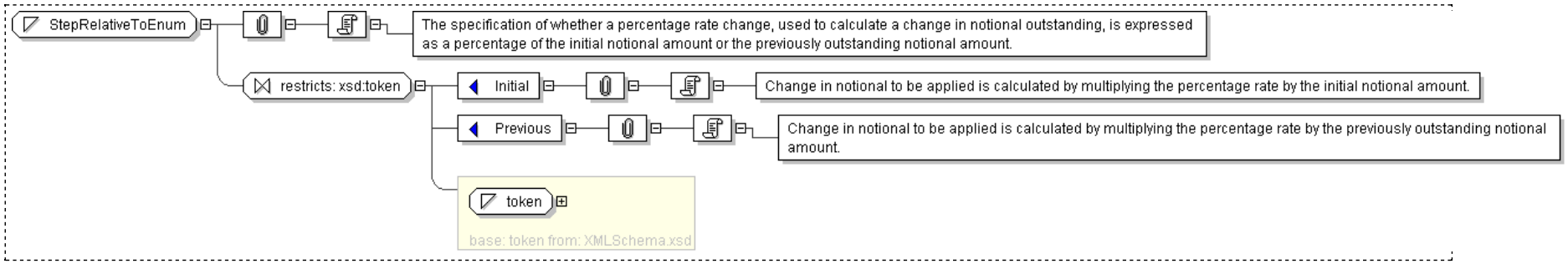
Simple Type: StepRelativeToEnum

[Table of contents]

Super-types:	xsd:token < StepRelativeToEnum (by restriction)
Sub-types:	None

Name	StepRelativeToEnum
Content	• Base XSD Type: token • *value* comes from list: {'Initial'|'Previous'}
Documentation	The specification of whether a percentage rate change, used to calculate a change in notional outstanding, is expressed as a percentage of the initial notional amount or the previously outstanding notional amount.

Diagram



Schema Component Representation

```
<xsd:simpleType name="StepRelativeToEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Initial"/>
    <xsd:enumeration value="Previous"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

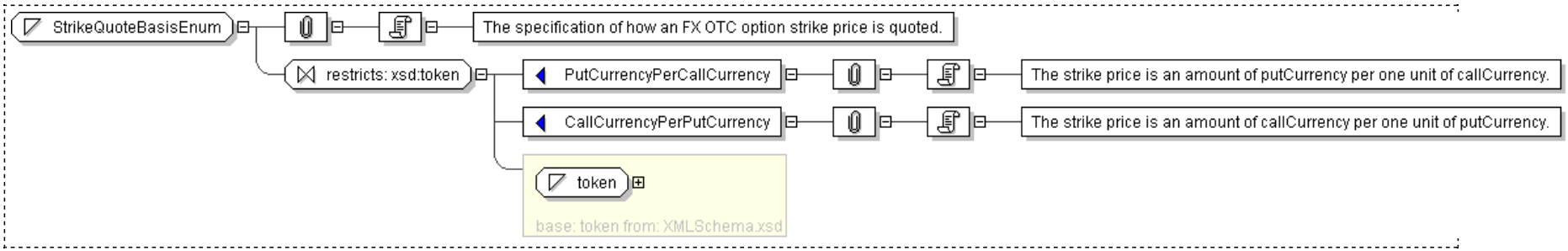
Simple Type: StrikeQuoteBasisEnum

[Table of contents]

Super-types:	xsd:token < StrikeQuoteBasisEnum (by restriction)
Sub-types:	None

Name	StrikeQuoteBasisEnum
Content	• Base XSD Type: token • *value* comes from list: {'PutCurrencyPerCallCurrency'|'CallCurrencyPerPutCurrency'}
Documentation	The specification of how an FX OTC option strike price is quoted.

Diagram



Schema Component Representation

```
<xsd:simpleType name="StrikeQuoteBasisEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="PutCurrencyPerCallCurrency"/>
    <xsd:enumeration value="CallCurrencyPerPutCurrency"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

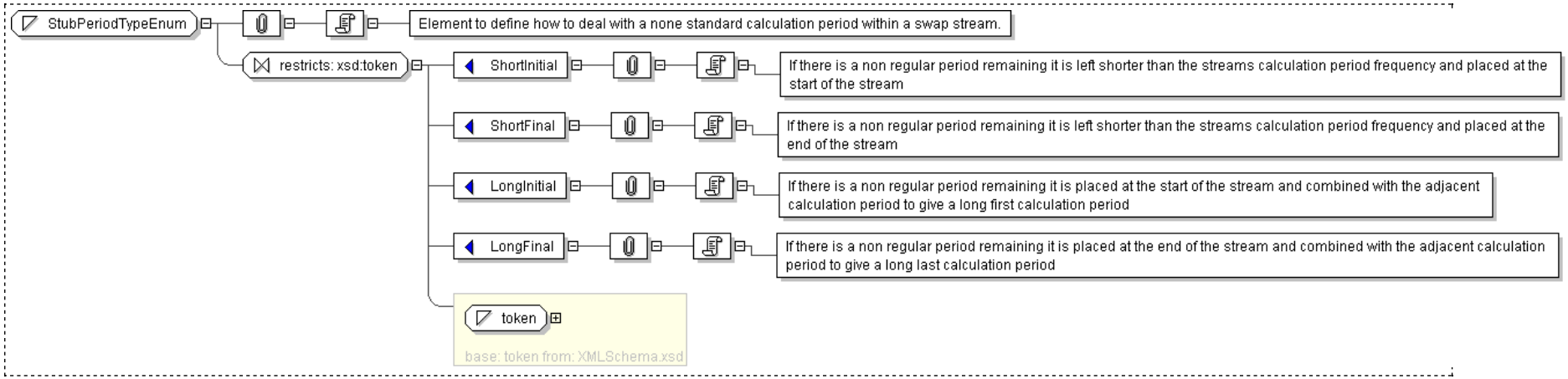
Simple Type: StubPeriodTypeEnum

[Table of contents]

Super-types:	xsd:token < StubPeriodTypeEnum (by restriction)
Sub-types:	None

Name	StubPeriodTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'ShortInitial'|'ShortFinal'|'LongInitial'|'LongFinal'}
Documentation	Element to define how to deal with a none standard calculation period within a swap stream.

Diagram



Schema Component Representation

```
<xsd:simpleType name="StubPeriodTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="ShortInitial"/>
    <xsd:enumeration value="ShortFinal"/>
    <xsd:enumeration value="LongInitial"/>
    <xsd:enumeration value="LongFinal"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

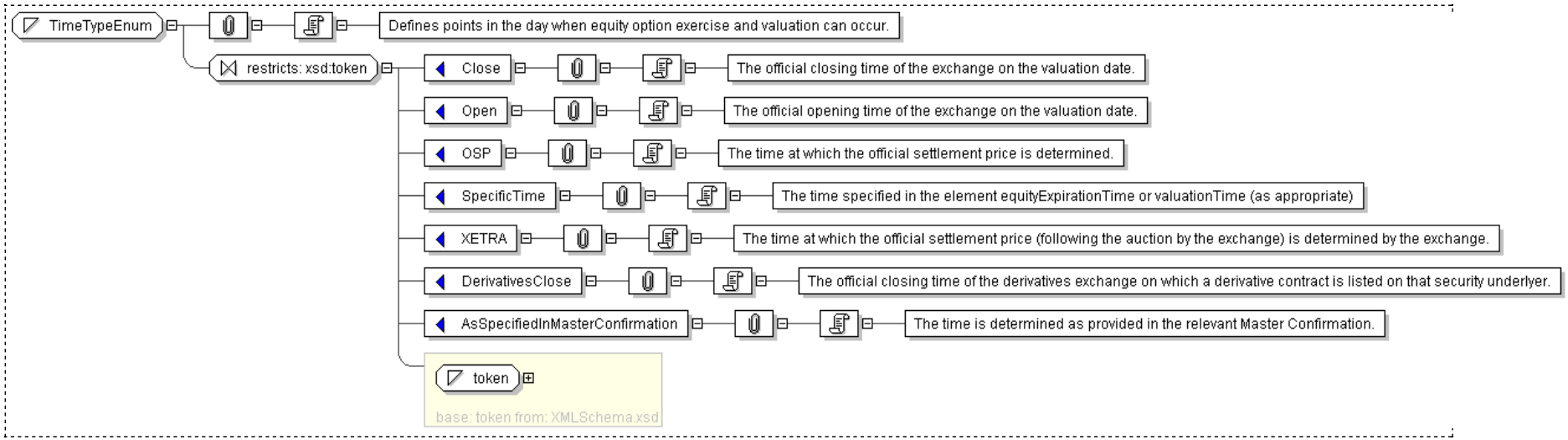
Simple Type: TimeTypeEnum

[Table of contents]

Super-types:	xsd:token < TimeTypeEnum (by restriction)
Sub-types:	None

Name	TimeTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Close'|'Open'|'OSP'|'SpecificTime'|'XETRA'|'DerivativesClose'|'AsSpecifiedInMasterConfirmation'}
Documentation	Defines points in the day when equity option exercise and valuation can occur.

Diagram



Schema Component Representation

```
<xsd:simpleType name="TimeTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Close"/>
    <xsd:enumeration value="Open"/>
    <xsd:enumeration value="OSP"/>
    <xsd:enumeration value="SpecificTime"/>
    <xsd:enumeration value="XETRA"/>
    <xsd:enumeration value="DerivativesClose"/>
    <xsd:enumeration value="AsSpecifiedInMasterConfirmation"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

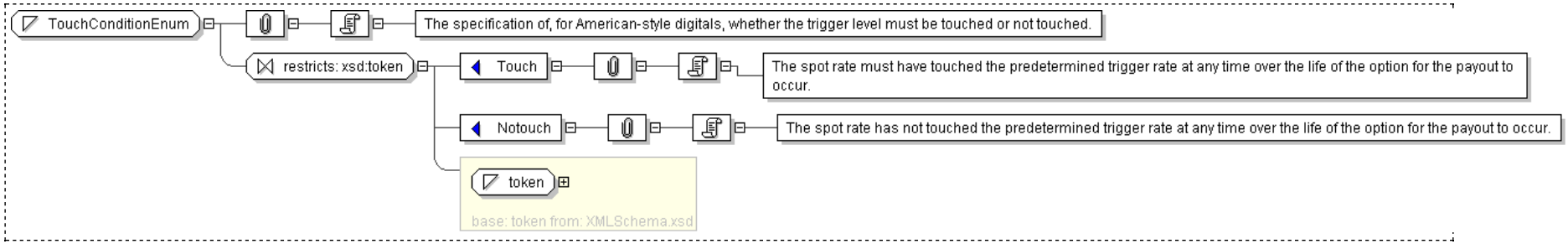
Simple Type: TouchConditionEnum

[Table of contents]

Super-types:	xsd:token < TouchConditionEnum (by restriction)
Sub-types:	None

Name	TouchConditionEnum
Content	• Base XSD Type: token • *value* comes from list: {'Touch'|'Notouch'}
Documentation	The specification of, for American-style digitals, whether the trigger level must be touched or not touched.

Diagram



Schema Component Representation

```
<xsd:simpleType name="TouchConditionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Touch"/>
    <xsd:enumeration value="Notouch"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

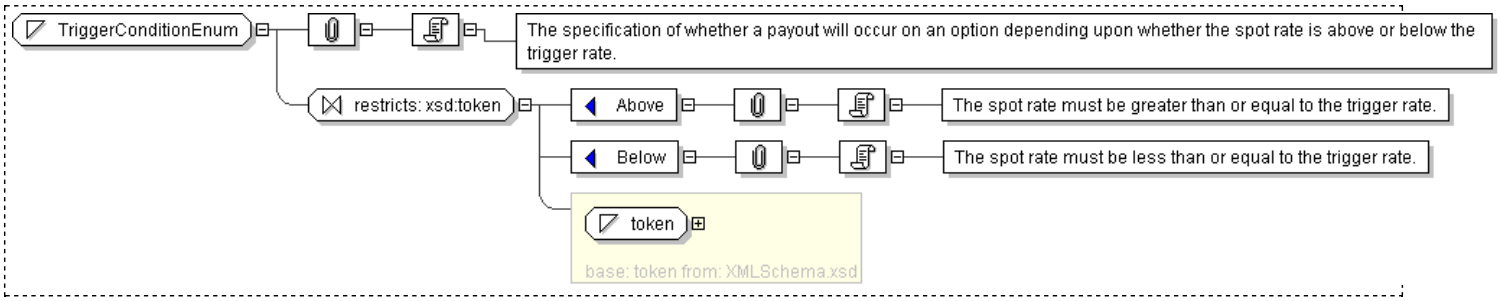
Simple Type: TriggerConditionEnum

[Table of contents]

Super-types:	xsd:token < TriggerConditionEnum (by restriction)
Sub-types:	None

Name	TriggerConditionEnum
Content	• Base XSD Type: token • *value* comes from list: {'Above'|'Below'}
Documentation	The specification of whether a payout will occur on an option depending upon whether the spot rate is above or below the trigger rate.

Diagram



Schema Component Representation

```
<xsd:simpleType name="TriggerConditionEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Above"/>
    <xsd:enumeration value="Below"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

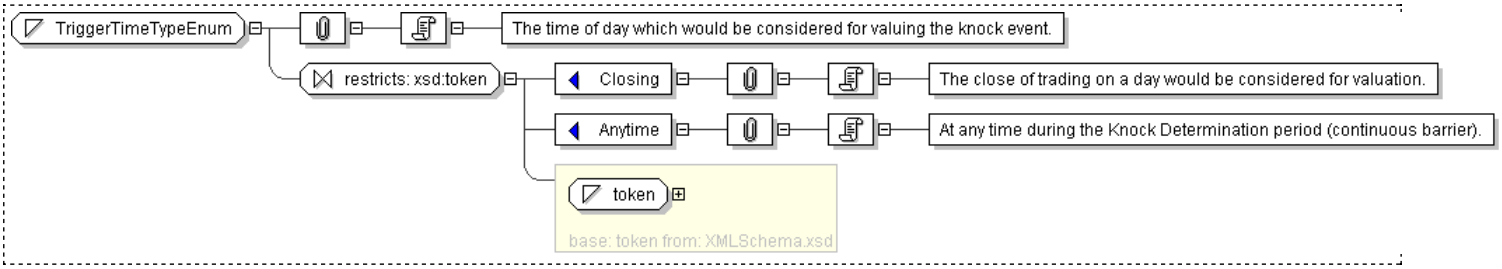
Simple Type: TriggerTimeTypeEnum

[Table of contents]

Super-types:	xsd:token < TriggerTimeTypeEnum (by restriction)
Sub-types:	None

Name	TriggerTimeTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'Closing'|'Anytime'}
Documentation	The time of day which would be considered for valuing the knock event.

Diagram



Schema Component Representation

```
<xsd:simpleType name="TriggerTimeTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Closing"/>
    <xsd:enumeration value="Anytime"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

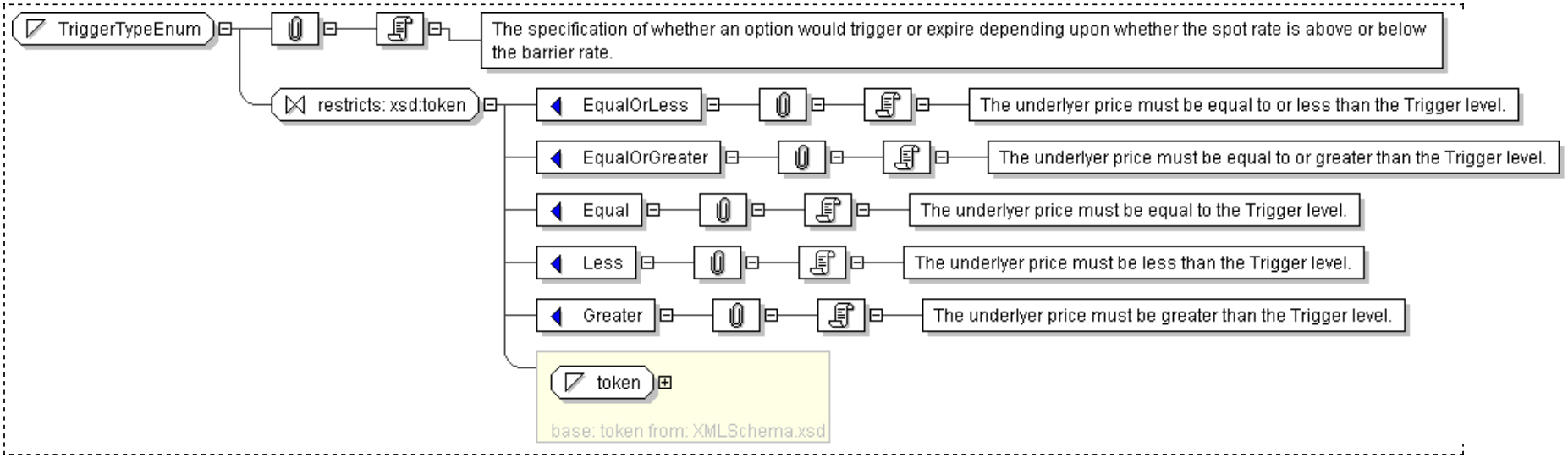
Simple Type: TriggerTypeEnum

[Table of contents]

Super-types:	xsd:token < TriggerTypeEnum (by restriction)
Sub-types:	None

Name	TriggerTypeEnum
Content	• Base XSD Type: token • *value* comes from list: {'EqualOrLess'|'EqualOrGreater'|'Equal'|'Less'|'Greater'}
Documentation	The specification of whether an option would trigger or expire depending upon whether the spot rate is above or below the barrier rate.

Diagram



Schema Component Representation

```
<xsd:simpleType name="TriggerTypeEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="EqualOrLess"/>
    <xsd:enumeration value="EqualOrGreater"/>
    <xsd:enumeration value="Equal"/>
    <xsd:enumeration value="Less"/>
    <xsd:enumeration value="Greater"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

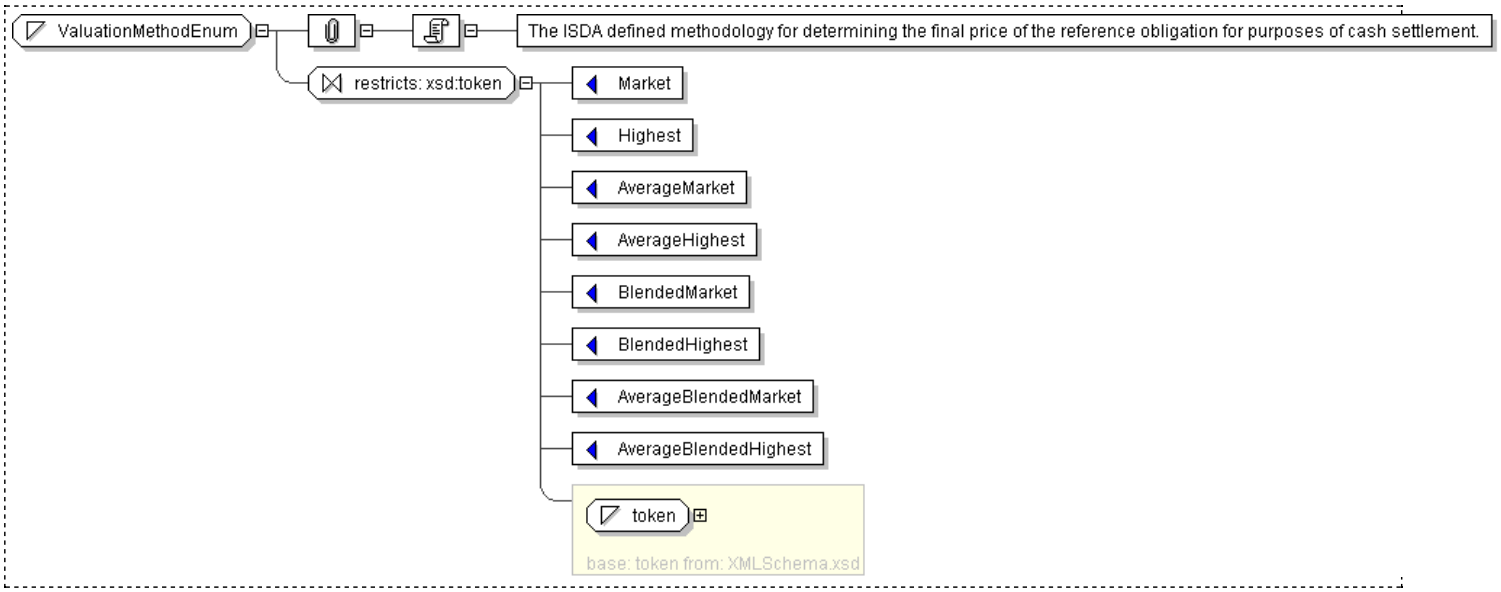
Simple Type: ValuationMethodEnum

[Table of contents]

Super-types:	xsd:token < ValuationMethodEnum (by restriction)
Sub-types:	None

Name	ValuationMethodEnum
Content	• Base XSD Type: token • *value* comes from list: {'Market'|'Highest'|'AverageMarket'|'AverageHighest'|'BlendedMarket'|'BlendedHighest'|'AverageBlendedMarket'|'AverageBlendedHighest'}
Documentation	The ISDA defined methodology for determining the final price of the reference obligation for purposes of cash settlement.

Diagram



Schema Component Representation

```
<xsd:simpleType name="ValuationMethodEnum">
  <xsd:restriction base=" xsd:token ">
    <xsd:enumeration value="Market"/>
    <xsd:enumeration value="Highest"/>
    <xsd:enumeration value="AverageMarket"/>
    <xsd:enumeration value="AverageHighest"/>
    <xsd:enumeration value="BlendedMarket"/>
    <xsd:enumeration value="BlendedHighest"/>
    <xsd:enumeration value="AverageBlendedMarket"/>
    <xsd:enumeration value="AverageBlendedHighest"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Simple Type: WeeklyRollConventionEnum

[Table of contents]

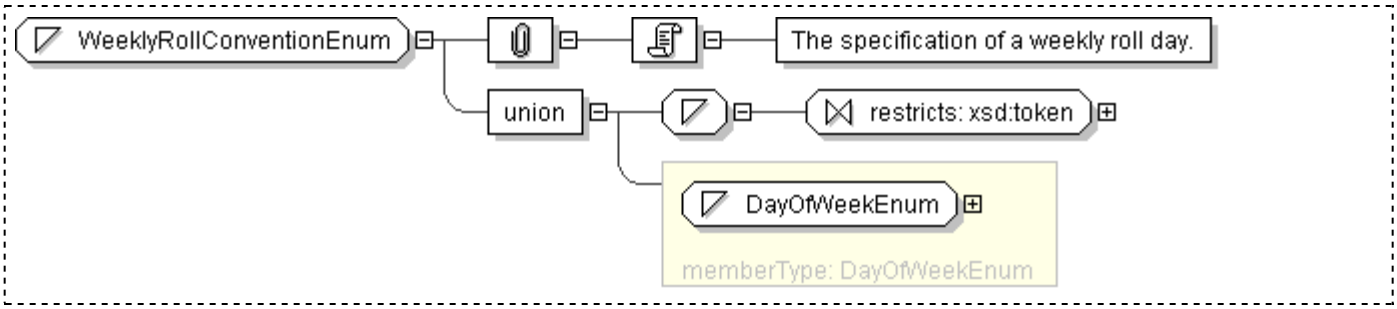
Super-types:	None
Sub-types:	None

Name	WeeklyRollConventionEnum

Content	• Union of following types:
	  ◦ DayOfWeekEnum
	  ◦ Locally defined type:
	  ▪ Base XSD Type: token
	  ▪ *value* comes from list: {'TBILL'}

Documentation	The specification of a weekly roll day.

Diagram



Schema Component Representation

```
<xsd:simpleType name="WeeklyRollConventionEnum">
   <xsd:union memberTypes=" DayOfWeekEnum ">
      <xsd:simpleType>
         <xsd:restriction base=" xsd:token ">
            <xsd:enumeration value="TBILL"/>
         </xsd:restriction>
      </xsd:simpleType>
   </xsd:union>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: brokerEquityOption
  - Element: equityForward
  - Element: equityOption
  - Element: equityOptionTransactionSupplement
- Global Definitions
  - Complex Type: BrokerEquityOption
  - Complex Type: EquityAmericanExercise
  - Complex Type: EquityBermudaExercise
  - Complex Type: EquityDerivativeBase
  - Complex Type: EquityDerivativeLongFormBase
  - Complex Type: EquityDerivativeShortFormBase
  - Complex Type: EquityEuropeanExercise
  - Complex Type: EquityExerciseValuationSettlement
  - Complex Type: EquityForward
  - Complex Type: EquityMultipleExercise
  - Complex Type: EquityOption
  - Complex Type: EquityOptionTermination
  - Complex Type: EquityOptionTransactionSupplement
  - Complex Type: PrePayment
  - Model Group: EquityExpiration.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-eq-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation

xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-eq-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: brokerEquityOption

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

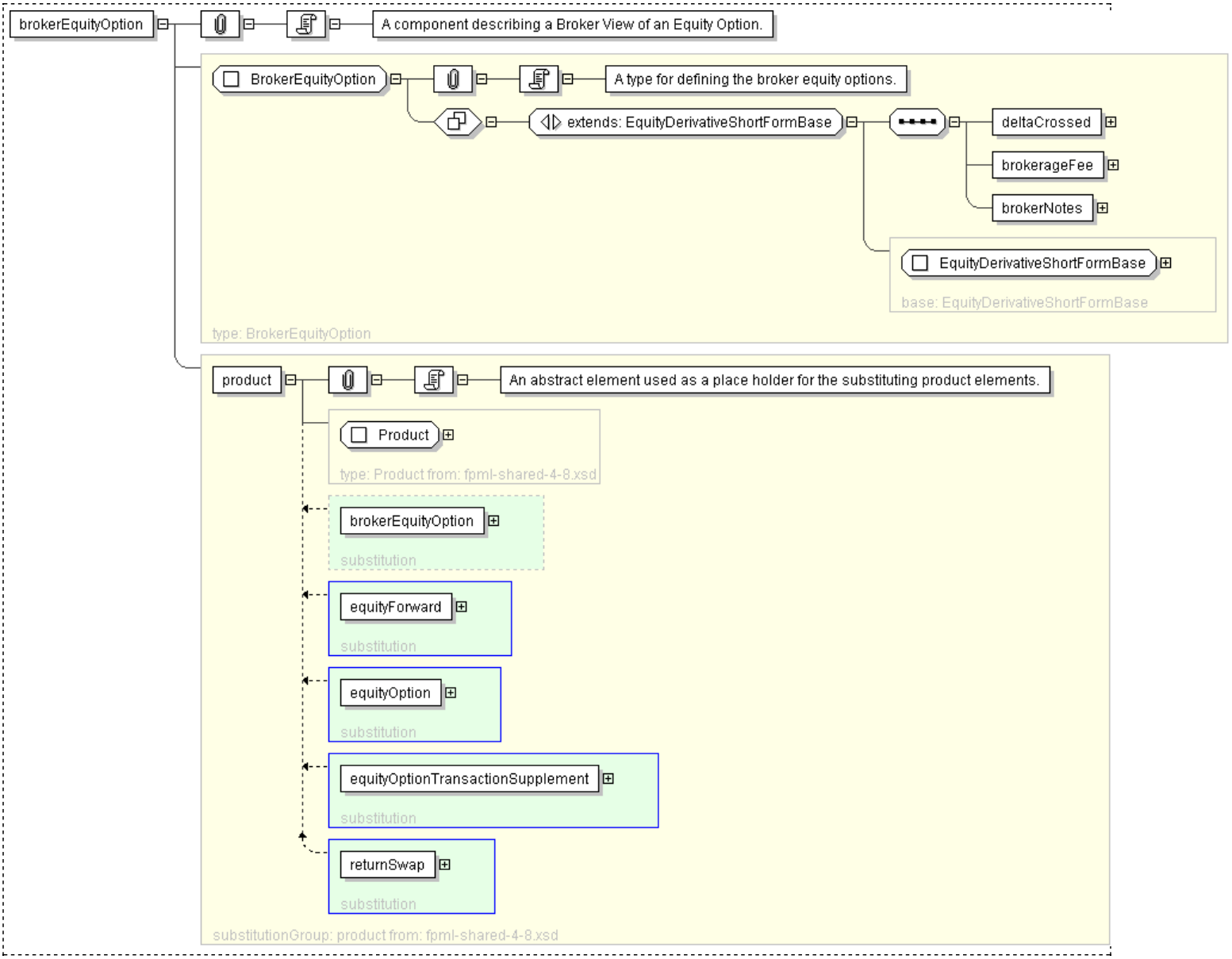
Name	brokerEquityOption
Type	BrokerEquityOption
Nillable	no
Abstract	no
Documentation	A component describing a Broker View of an Equity Option.

Logical Diagram

XML Instance Representation

```
<brokerEquityOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the
   case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in either equity, index or
   convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued and how it is settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
      'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
      'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <strike> EquityStrike </strike> [1]
   'Defines whether it is a price or level at which the option has been, or will be, struck.'

   <spotPrice> NonNegativeDecimal </spotPrice> [0..1]
   'The price per share, index or basket observed on the trade or effective date.'

   <numberOfOptions> NonNegativeDecimal </numberOfOptions> [1]
   'The number of options comprised in the option transaction.'

   <equityPremium> EquityPremium </equityPremium> [1]
   'The equity option premium payable by the buyer to the seller.'

   <deltaCrossed> xsd:boolean </deltaCrossed> [1]
   <brokerageFee> Money </brokerageFee> [1]
   <brokerNotes> xsd:string </brokerNotes> [1]
</brokerEquityOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="brokerEquityOption" type=" BrokerEquityOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

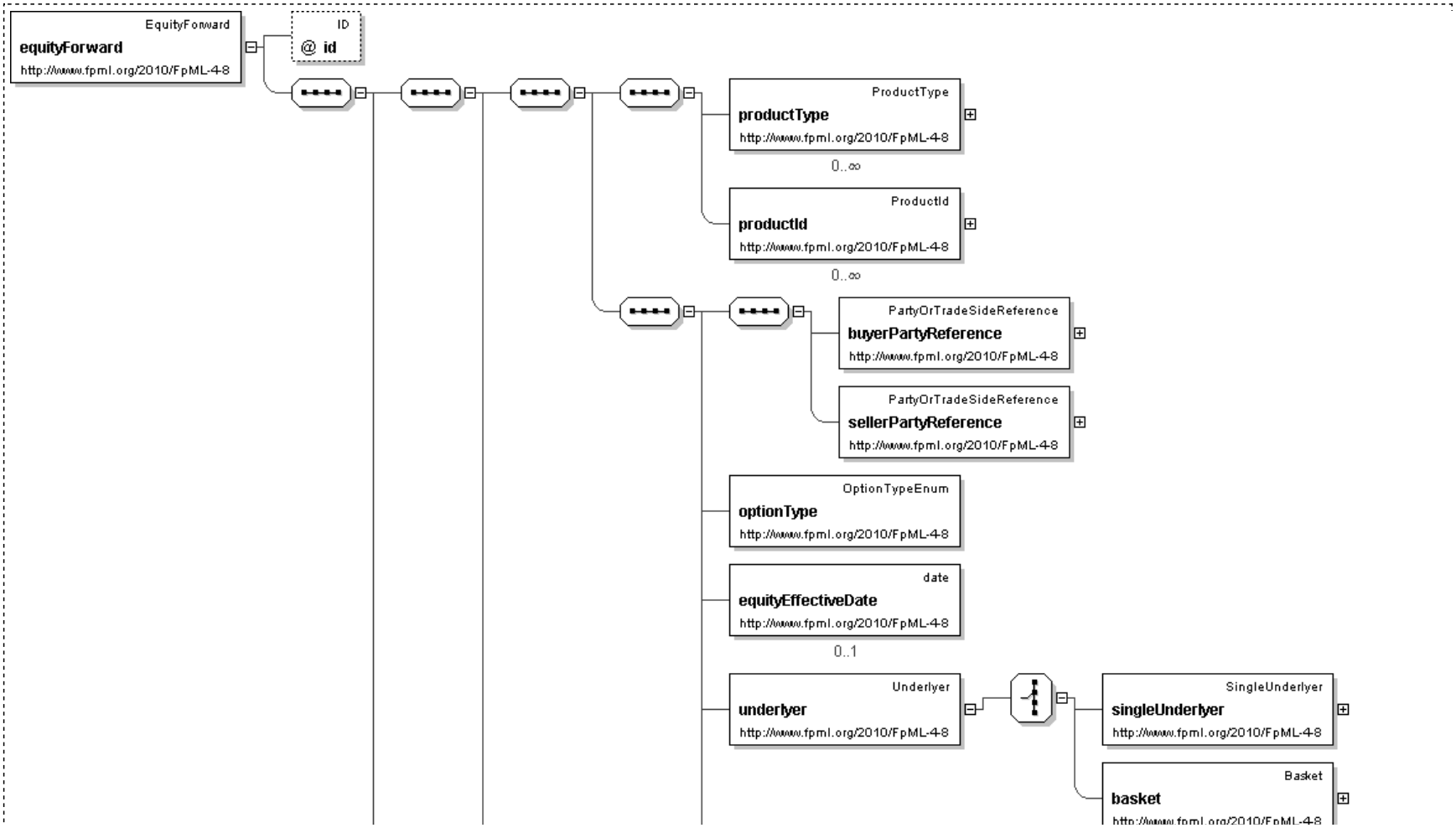
Element: equityForward

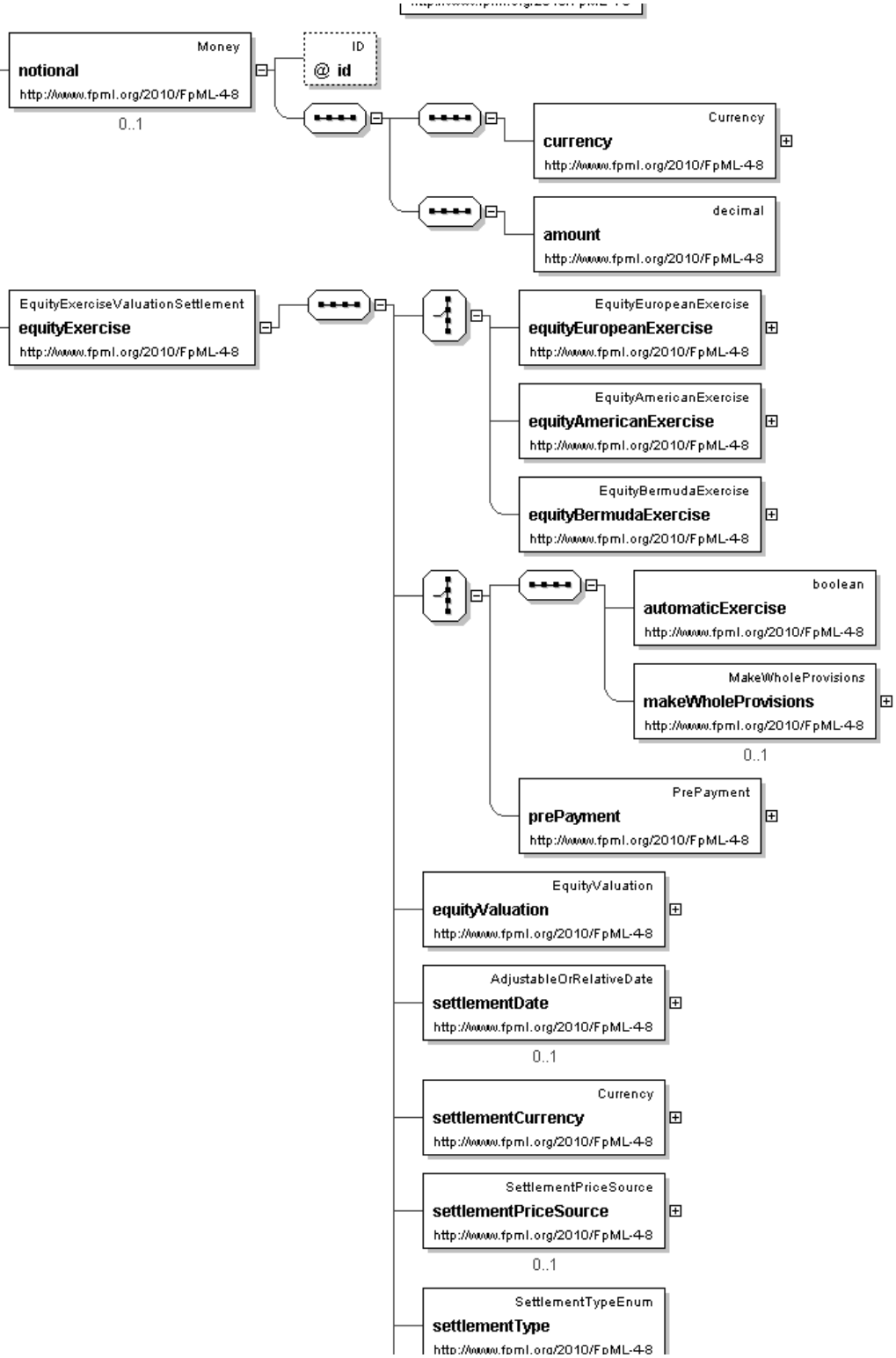
[Table of contents]

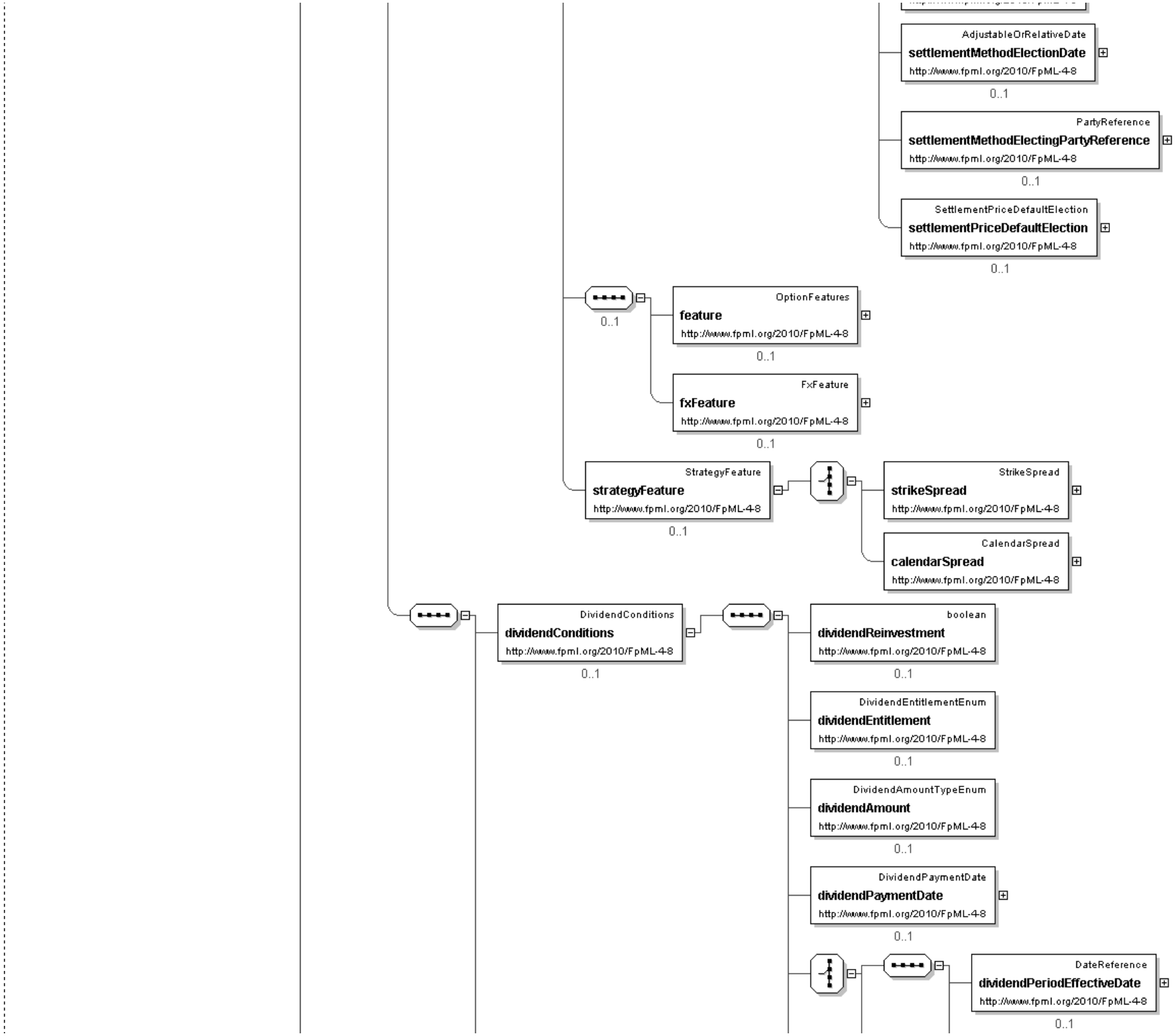
- *This element can be used wherever the following element is referenced:*
  - product

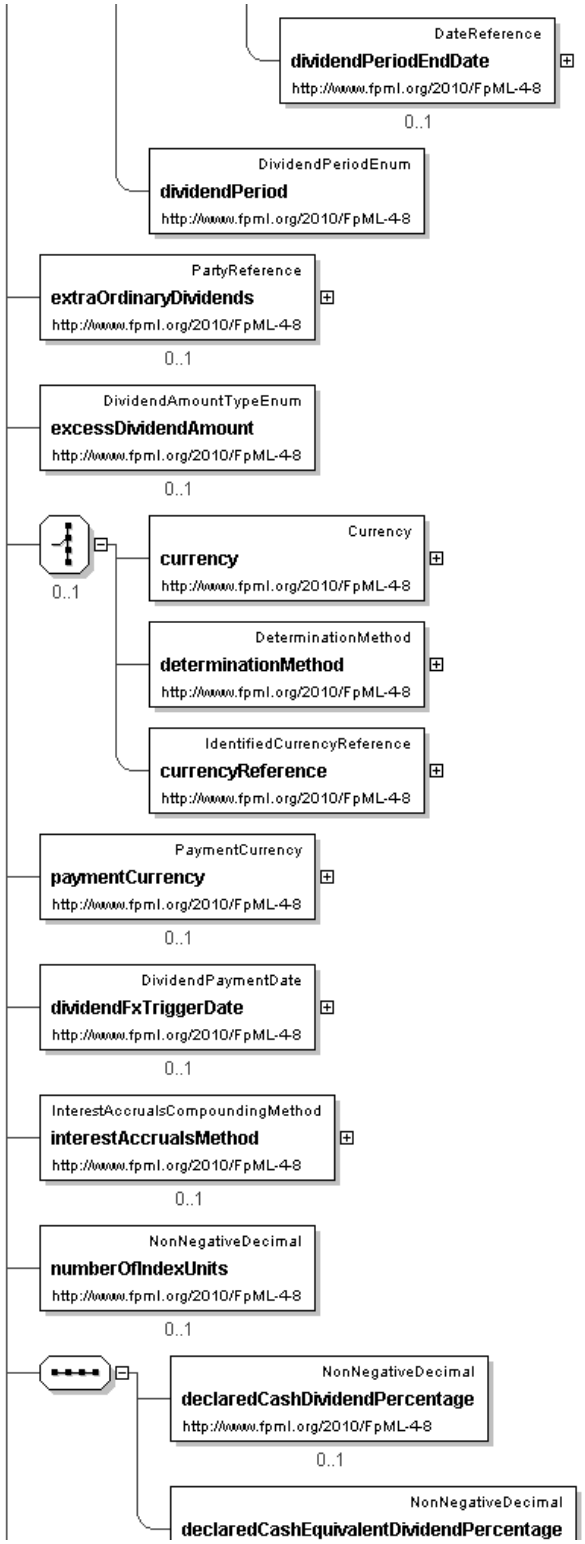
Name	equityForward
Type	EquityForward
Nillable	no
Abstract	no
Documentation	A component describing an Equity Forward product.

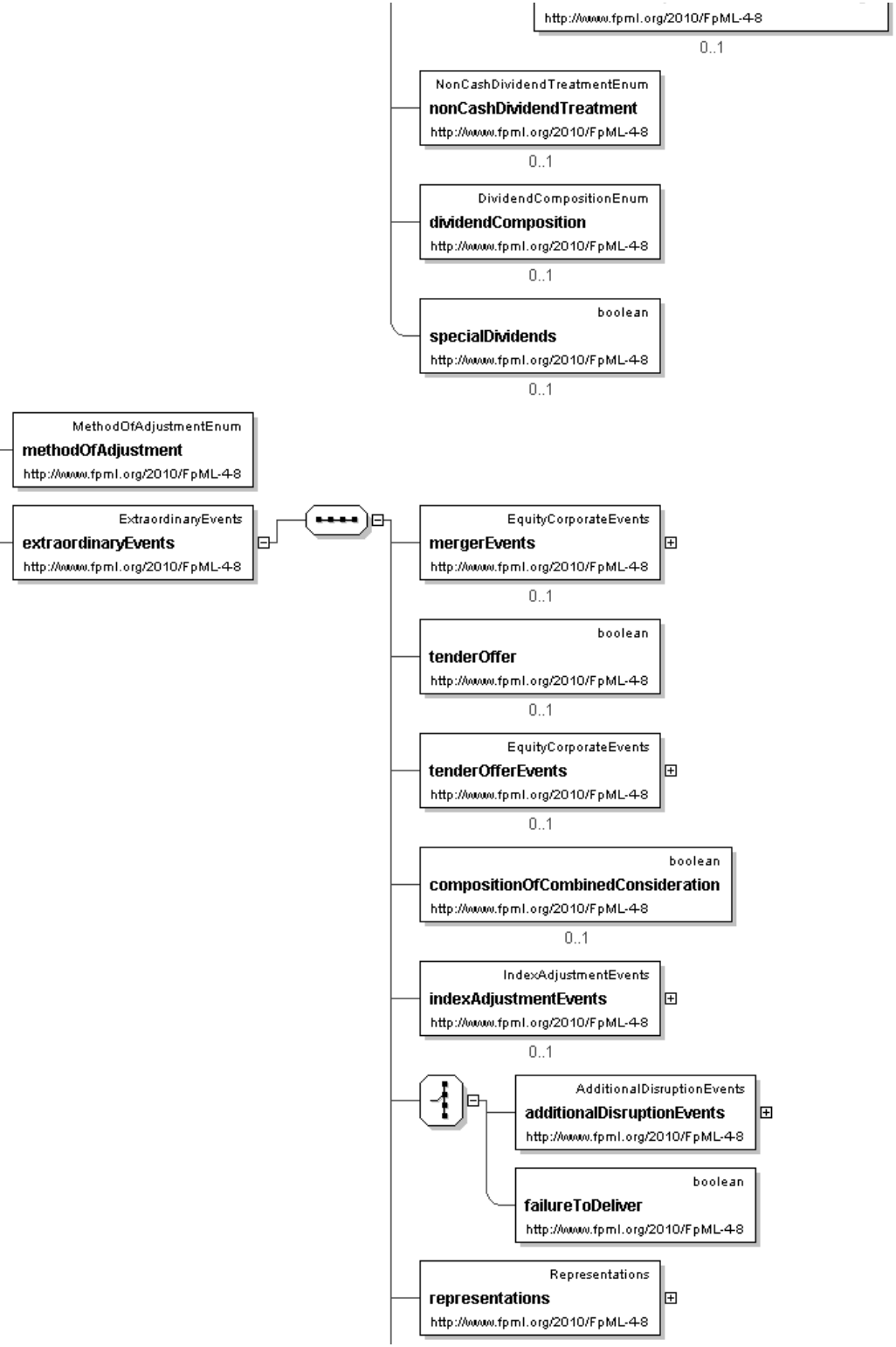
Logical Diagram

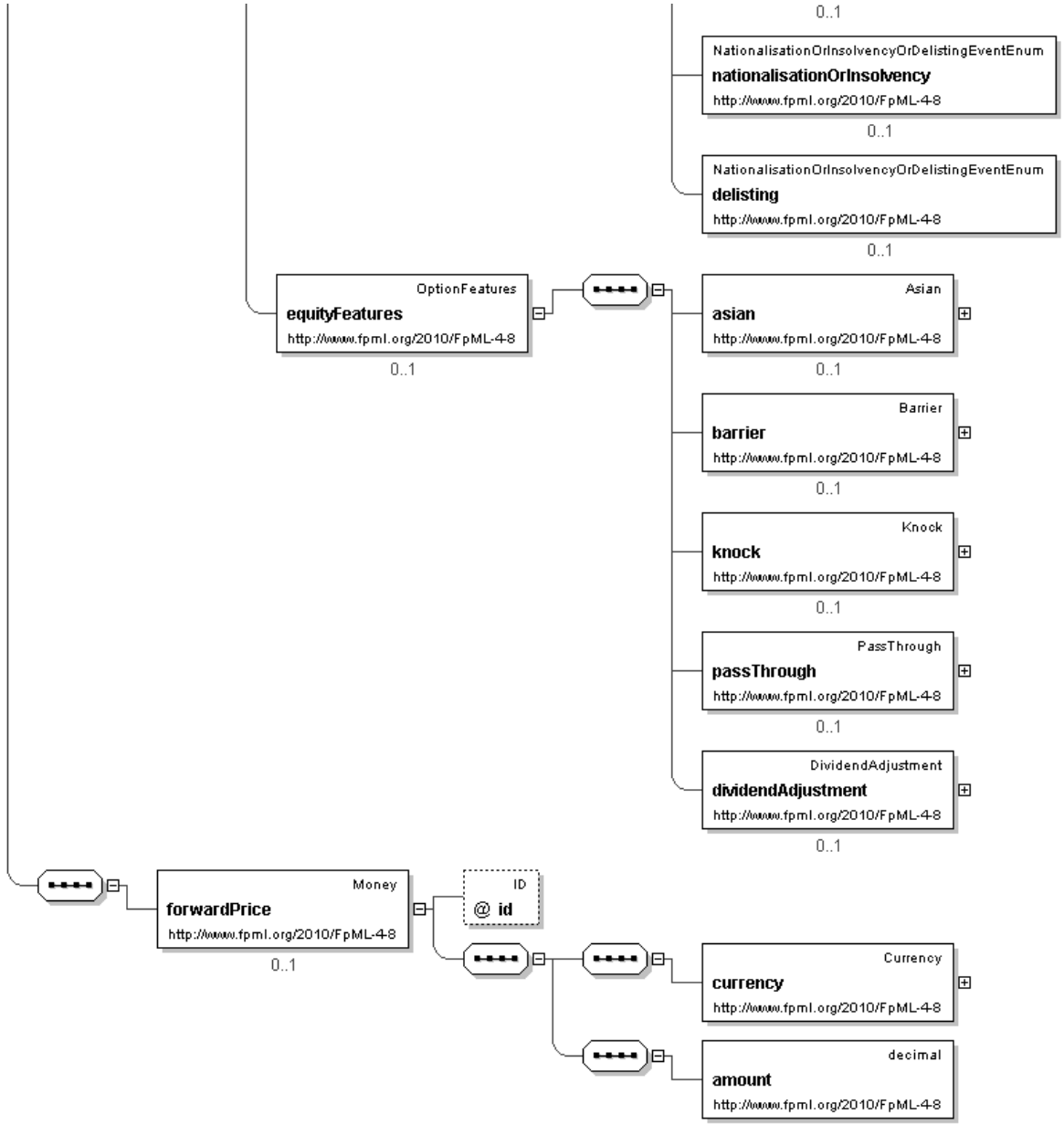



Money
notional
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
EquityExerciseValuationSettlement
equityExercise
http://www.fpml.org/2010/FpML-4-8
EquityEuropeanExercise
equityEuropeanExercise
http://www.fpml.org/2010/FpML-4-8
EquityAmericanExercise
equityAmericanExercise
http://www.fpml.org/2010/FpML-4-8
EquityBermudaExercise
equityBermudaExercise
http://www.fpml.org/2010/FpML-4-8
boolean
automaticExercise
http://www.fpml.org/2010/FpML-4-8
MakeWholeProvisions
makeWholeProvisions
http://www.fpml.org/2010/FpML-4-8
0..1
PrePayment
prePayment
http://www.fpml.org/2010/FpML-4-8
EquityValuation
equityValuation
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
settlementDate
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
SettlementPriceSource
settlementPriceSource
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementTypeEnum
settlementType


AdjustableOrRelativeDate
settlementMethodElectionDate
http://www.fpml.org/2010/FpML-4-8
0..1
PartyReference
settlementMethodElectingPartyReference
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementPriceDefaultElection
settlementPriceDefaultElection
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
OptionFeatures
feature
http://www.fpml.org/2010/FpML-4-8
0..1
FxFeature
fxFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrategyFeature
strategyFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrikeSpread
strikeSpread
http://www.fpml.org/2010/FpML-4-8
CalendarSpread
calendarSpread
http://www.fpml.org/2010/FpML-4-8
DividendConditions
dividendConditions
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
dividendReinvestment
http://www.fpml.org/2010/FpML-4-8
0..1
DividendEntitlementEnum
dividendEntitlement
http://www.fpml.org/2010/FpML-4-8
0..1
DividendAmountTypeEnum
dividendAmount
http://www.fpml.org/2010/FpML-4-8
0..1
DividendPaymentDate
dividendPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
DateReference
dividendPeriodEffectiveDate
http://www.fpml.org/2010/FpML-4-8
0..1


DateReference
dividendPeriodEndDate
http://www.fpml.org/2010/FpML-4-8
0..1
DividendPeriodEnum
dividendPeriod
http://www.fpml.org/2010/FpML-4-8
PartyReference
extraOrdinaryDividends
http://www.fpml.org/2010/FpML-4-8
0..1
DividendAmountTypeEnum
excessDividendAmount
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2010/FpML-4-8
PaymentCurrency
paymentCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
DividendPaymentDate
dividendFxTriggerDate
http://www.fpml.org/2010/FpML-4-8
0..1
InterestAccrualsCompoundingMethod
interestAccrualsMethod
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
numberOfIndexUnits
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
declaredCashDividendPercentage
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
declaredCashEquivalentDividendPercentage


http://www.fpml.org/2010/FpML-4-8
0..1
NonCashDividendTreatmentEnum
nonCashDividendTreatment
http://www.fpml.org/2010/FpML-4-8
0..1
DividendCompositionEnum
dividendComposition
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
specialDividends
http://www.fpml.org/2010/FpML-4-8
0..1
MethodOfAdjustmentEnum
methodOfAdjustment
http://www.fpml.org/2010/FpML-4-8
ExtraordinaryEvents
extraordinaryEvents
http://www.fpml.org/2010/FpML-4-8
EquityCorporateEvents
mergerEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
tenderOffer
http://www.fpml.org/2010/FpML-4-8
0..1
EquityCorporateEvents
tenderOfferEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
compositionOfCombinedConsideration
http://www.fpml.org/2010/FpML-4-8
0..1
IndexAdjustmentEvents
indexAdjustmentEvents
http://www.fpml.org/2010/FpML-4-8
0..1
AdditionalDisruptionEvents
additionalDisruptionEvents
http://www.fpml.org/2010/FpML-4-8
boolean
failureToDeliver
http://www.fpml.org/2010/FpML-4-8
Representations
representations
http://www.fpml.org/2010/FpML-4-8



XML Instance Representation

```
<equityForward
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that
   the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA
   definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'
```

<sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

<optionType> OptionTypeEnum </optionType> [1]
'The type of option transaction.'

<equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
'Effective date for a forward starting option.'

<underlyer> Underlyer </underlyer> [1]
'Specifies the underlying component, which can be either one or many and consists in either equity, index or convertible bond component, or a combination of these.'

<notional> Money </notional> [0..1]
'The notional amount.'

<equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
'The parameters for defining how the equity option can be exercised, how it is valued and how it is settled.'

Start Group: Feature.model [0..1]
<feature> OptionFeatures </feature> [0..1]
'Asian, Barrier, Knock and Pass Through features.'

<fxFeature> FxFeature </fxFeature> [0..1]
'Quanto, Composite, or Cross Currency FX features.'

End Group: Feature.model
<strategyFeature> StrategyFeature </strategyFeature> [0..1]
'A equity option simple strategy feature.'

<dividendConditions> DividendConditions </dividendConditions> [0..1]
<methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [1]
'Defines how adjustments will be made to the contract should one or more of the extraordinary events occur.'

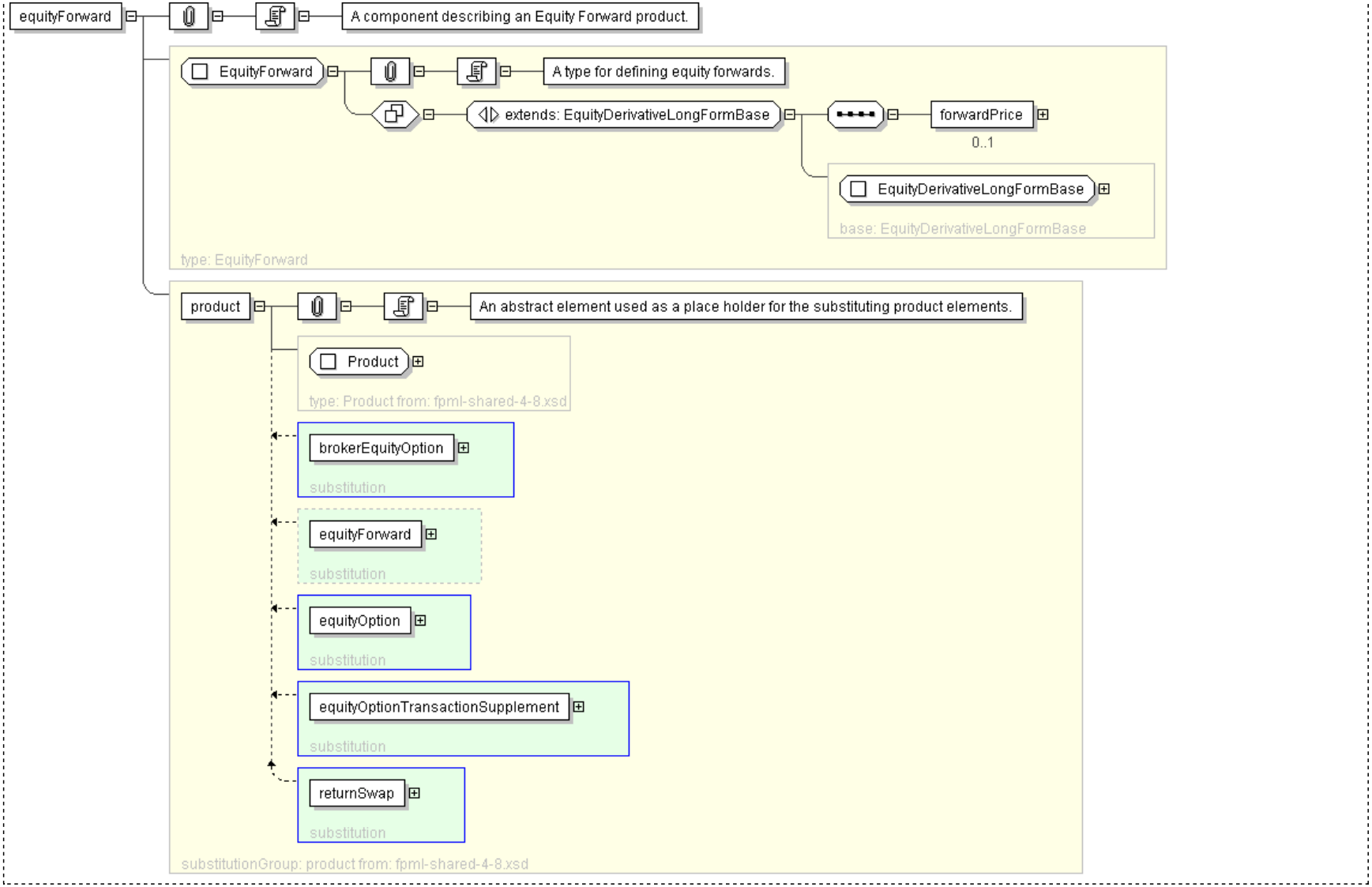
<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [1]
'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.'

<equityFeatures> OptionFeatures </equityFeatures> [0..1]
'DEPRECATED This element will be removed in the next FpML major version. Use the \"feature\" element for option features such as asian, barrier, knock.'

<forwardPrice> Money </forwardPrice> [0..1]
'The forward price per share, index or basket.'

</equityForward>

Diagram



Schema Component Representation

```
<xsd:element name="equityForward" type=" EquityForward " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: equityOption

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	equityOption
Type	EquityOption
Nillable	no
Abstract	no
Documentation	A component describing an Equity Option product.

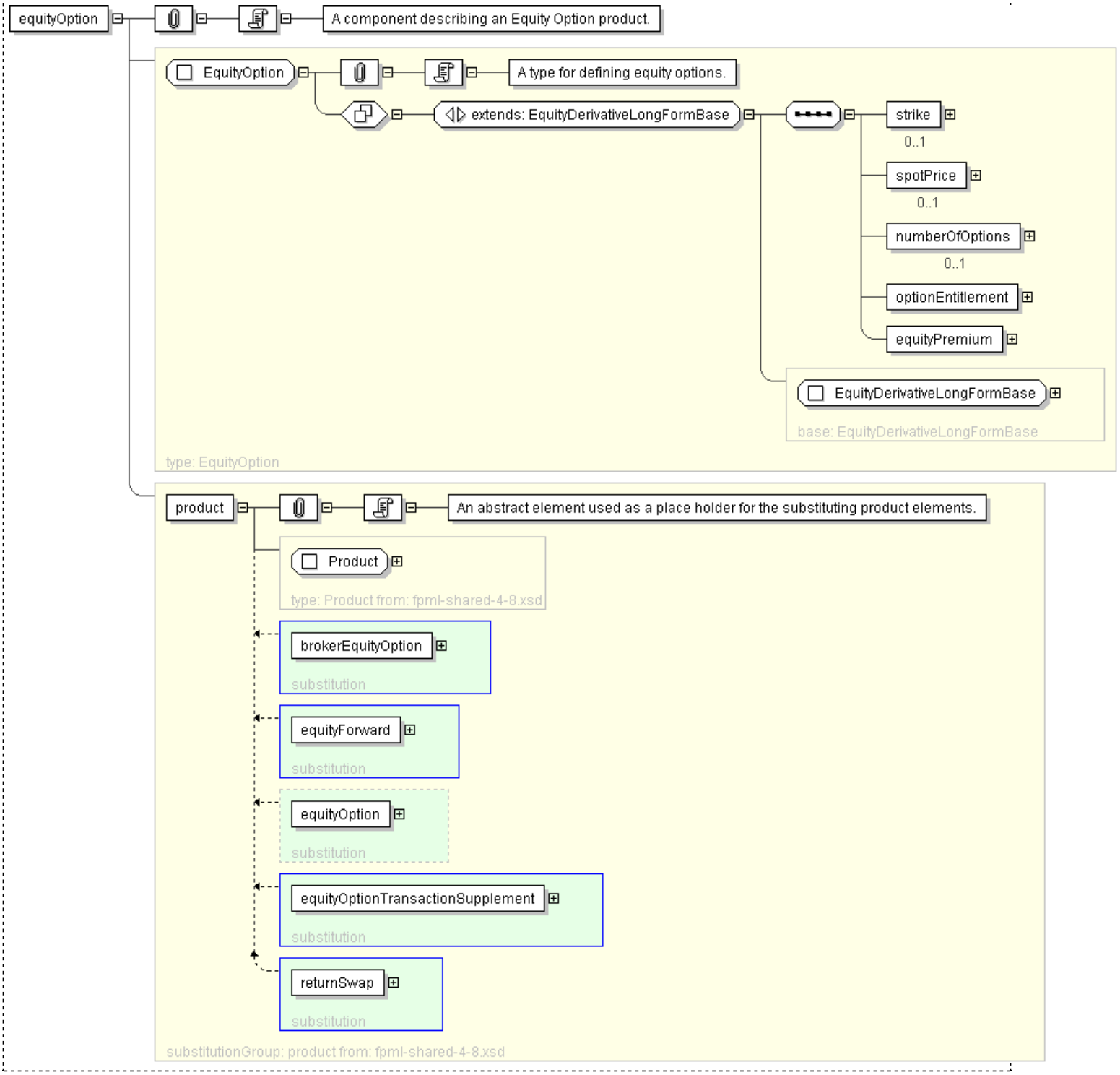
Logical Diagram

XML Instance Representation

```
<equityOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not strictly an
   enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the
   rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed
   rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights
   defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions
   Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in either equity,
   index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued and how it is
   settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
      'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
      'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <dividendConditions> DividendConditions </dividendConditions> [0..1]
   <methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [1]
   'Defines how adjustments will be made to the contract should one or more of the extraordinary events
   occur.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [1]
   'Where the underlying is shares, specifies events affecting the issuer of those shares that may
   require the terms of the transaction to be adjusted.'
```

```
<equityFeatures> OptionFeatures </equityFeatures> [0..1]
'DEPRECATED This element will be removed in the next FpML major version. Use the \"feature\" element
for option features such as asian, barrier, knock.'

<strike> EquityStrike </strike> [0..1]
'Defines whether it is a price or level at which the option has been, or will be, struck.'

<spotPrice> NonNegativeDecimal </spotPrice> [0..1]
'The price per share, index or basket observed on the trade or effective date.'

<numberOfOptions> NonNegativeDecimal </numberOfOptions> [0..1]
'The number of options comprised in the option transaction.'

<optionEntitlement> PositiveDecimal </optionEntitlement> [1]
'The number of shares per option comprised in the option transaction.'

<equityPremium> EquityPremium </equityPremium> [1]
'The equity option premium payable by the buyer to the seller.'

</equityOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="equityOption" type=" EquityOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

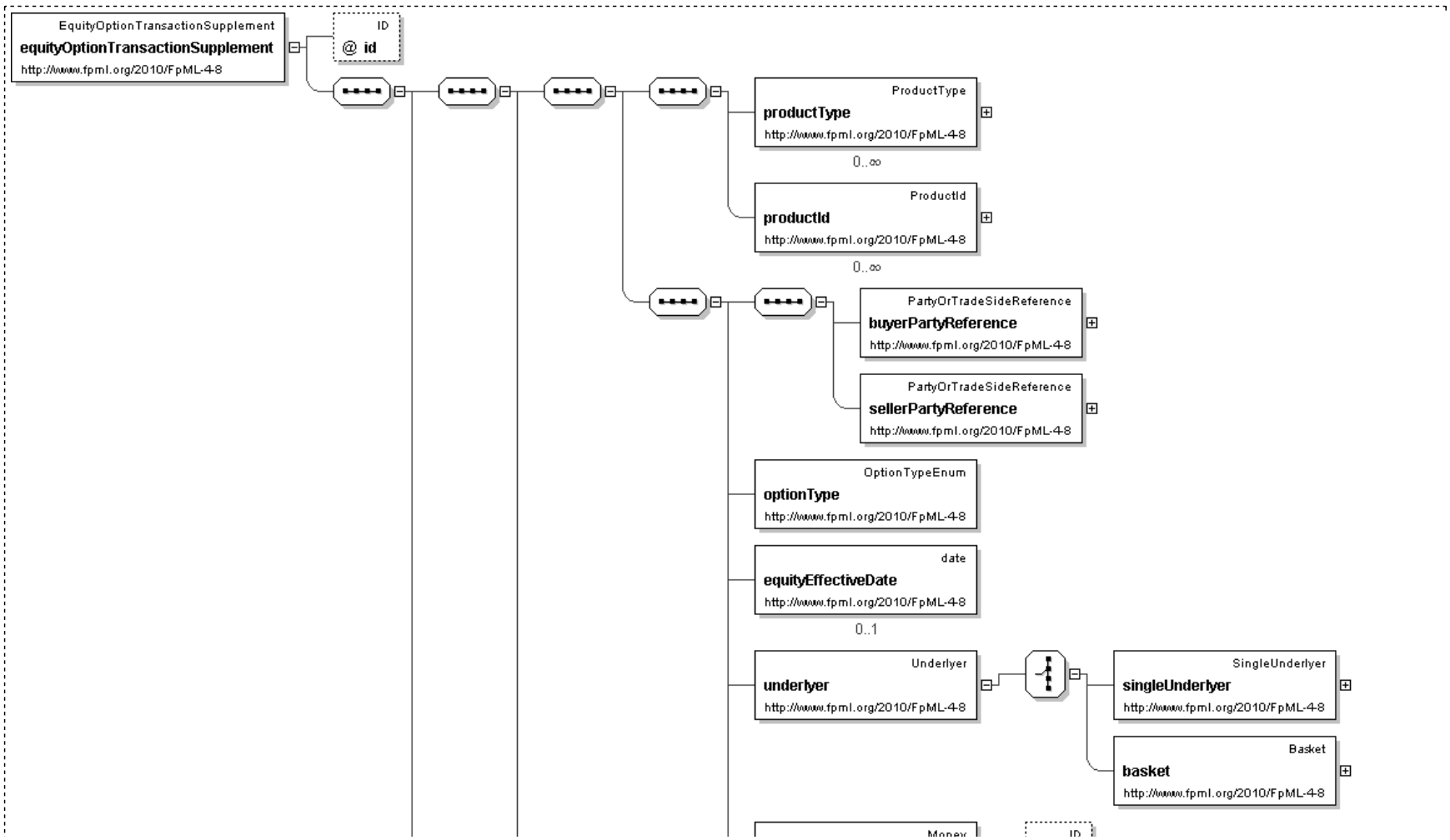
Element: equityOptionTransactionSupplement

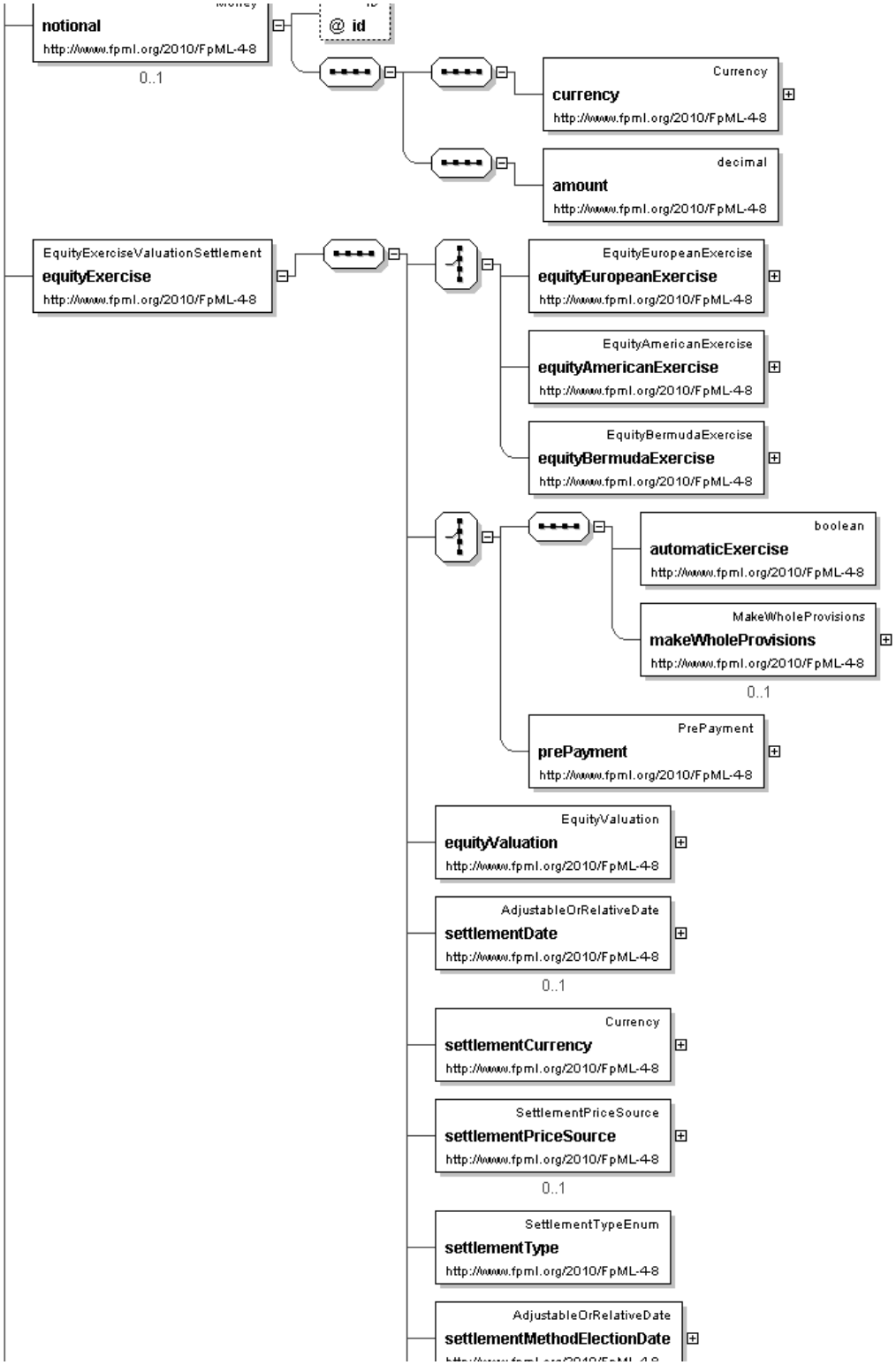
[Table of contents]

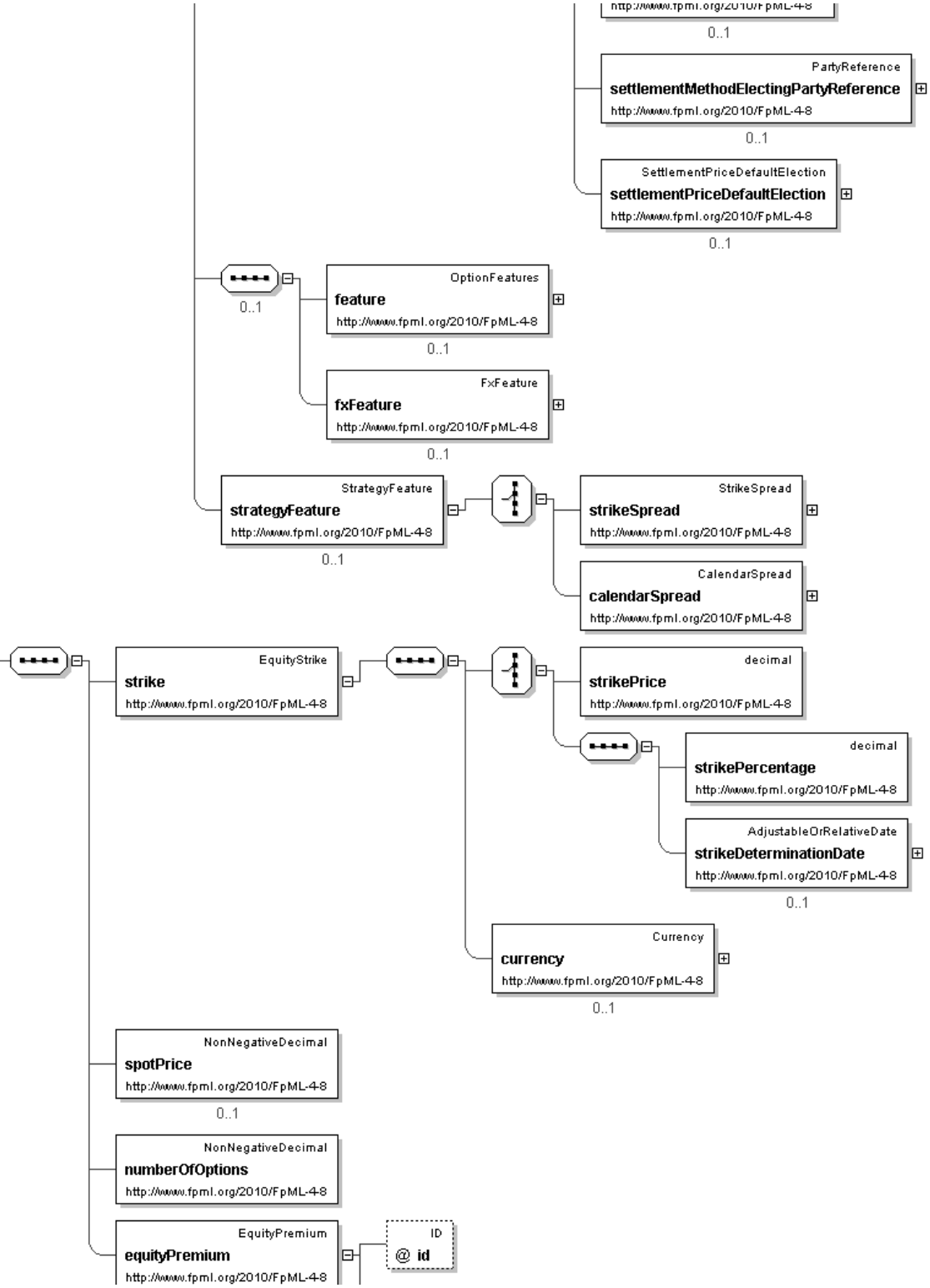
- *This element can be used wherever the following element is referenced:*
  - product

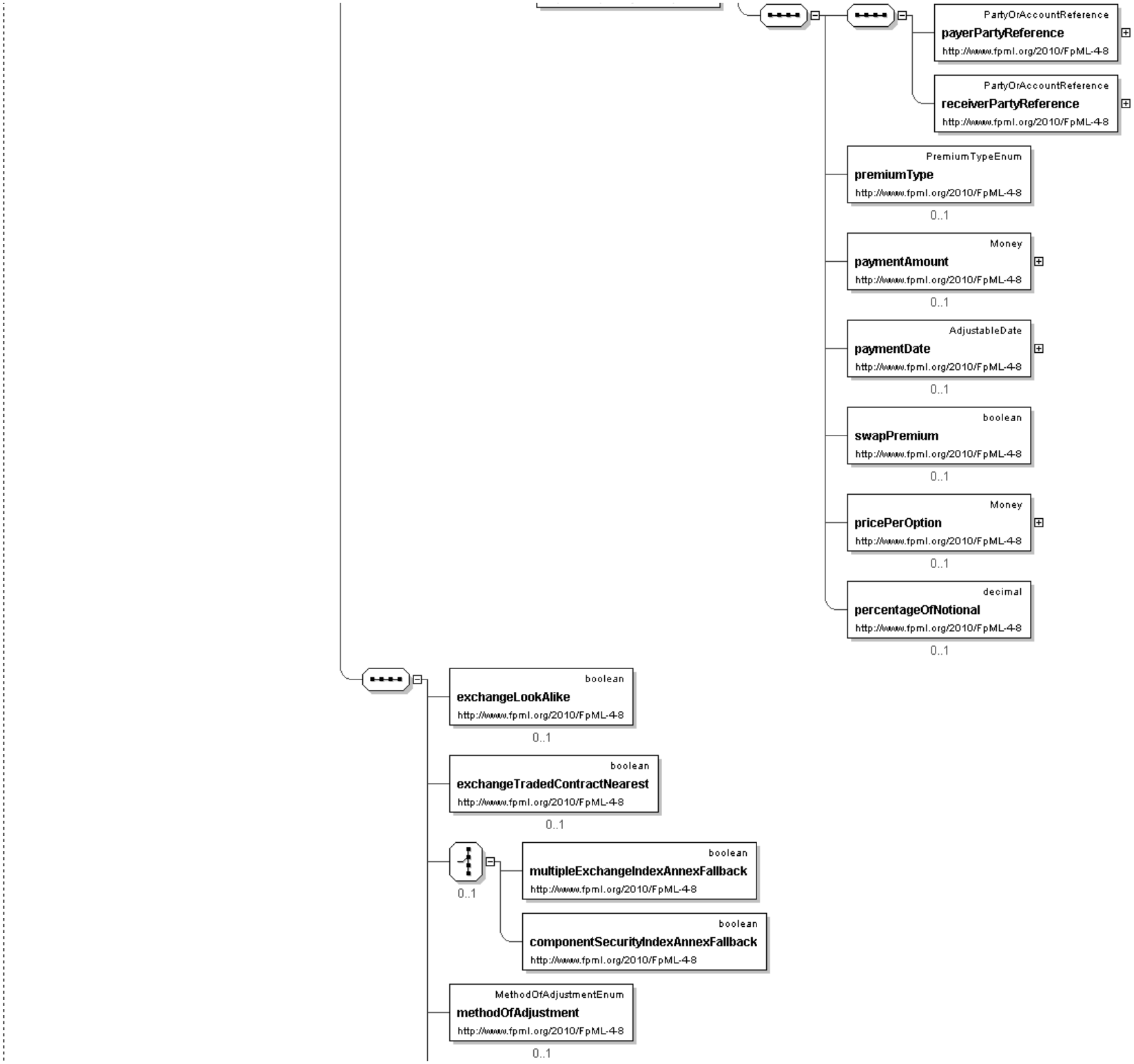
Name	equityOptionTransactionSupplement
Type	EquityOptionTransactionSupplement
Nillable	no
Abstract	no
Documentation	A component describing an Equity Option Transaction Supplement.

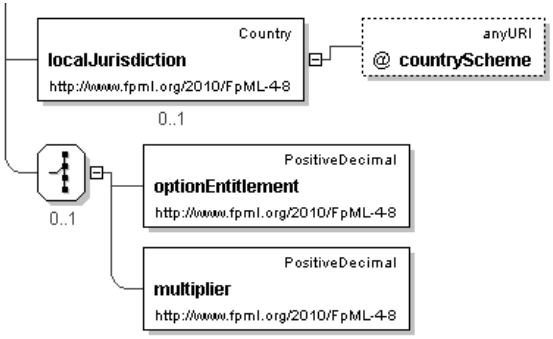
Logical Diagram




notional
http://www.fpml.org/2010/FpML-4-8
0..1
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
EquityExerciseValuationSettlement
equityExercise
http://www.fpml.org/2010/FpML-4-8
EquityEuropeanExercise
equityEuropeanExercise
http://www.fpml.org/2010/FpML-4-8
EquityAmericanExercise
equityAmericanExercise
http://www.fpml.org/2010/FpML-4-8
EquityBermudaExercise
equityBermudaExercise
http://www.fpml.org/2010/FpML-4-8
boolean
automaticExercise
http://www.fpml.org/2010/FpML-4-8
MakeWholeProvisions
makeWholeProvisions
http://www.fpml.org/2010/FpML-4-8
0..1
PrePayment
prePayment
http://www.fpml.org/2010/FpML-4-8
EquityValuation
equityValuation
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
settlementDate
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
SettlementPriceSource
settlementPriceSource
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementTypeEnum
settlementType
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
settlementMethodElectionDate


0..1
PartyReference
settlementMethodElectingPartyReference
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementPriceDefaultElection
settlementPriceDefaultElection
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
OptionFeatures
feature
http://www.fpml.org/2010/FpML-4-8
0..1
FxFeature
fxFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrategyFeature
strategyFeature
http://www.fpml.org/2010/FpML-4-8
0..1
StrikeSpread
strikeSpread
http://www.fpml.org/2010/FpML-4-8
CalendarSpread
calendarSpread
http://www.fpml.org/2010/FpML-4-8
EquityStrike
strike
http://www.fpml.org/2010/FpML-4-8
decimal
strikePrice
http://www.fpml.org/2010/FpML-4-8
decimal
strikePercentage
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
strikeDeterminationDate
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
spotPrice
http://www.fpml.org/2010/FpML-4-8
0..1
NonNegativeDecimal
numberOfOptions
http://www.fpml.org/2010/FpML-4-8
EquityPremium
equityPremium
http://www.fpml.org/2010/FpML-4-8
ID
@ id


PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
PremiumTypeEnum
premiumType
http://www.fpml.org/2010/FpML-4-8
0..1
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
0..1
AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
swapPremium
http://www.fpml.org/2010/FpML-4-8
0..1
Money
pricePerOption
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
percentageOfNotional
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
exchangeLookAlike
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
exchangeTradedContractNearest
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
boolean
multipleExchangeIndexAnnexFallback
http://www.fpml.org/2010/FpML-4-8
boolean
componentSecurityIndexAnnexFallback
http://www.fpml.org/2010/FpML-4-8
MethodOfAdjustmentEnum
methodOfAdjustment
http://www.fpml.org/2010/FpML-4-8
0..1

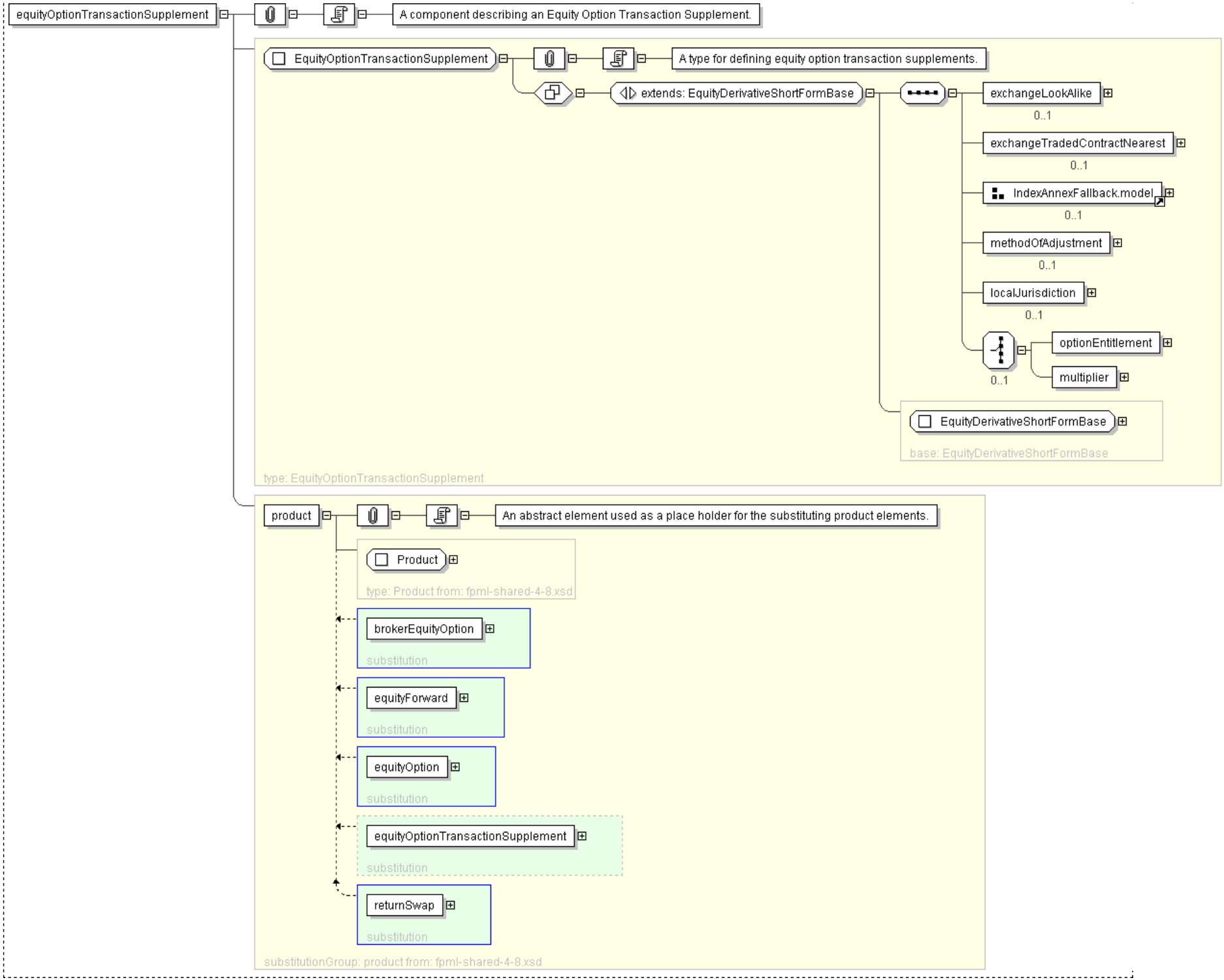


XML Instance Representation

```
<equityOptionTransactionSupplement
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the
    domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA
    definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return
    receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
    'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
    'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlying component, which can be either one or many and consists in either equity, index or convertible bond component, or
    a combination of these.'

   <notional> Money </notional> [0..1]
    'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
    'The parameters for defining how the equity option can be exercised, how it is valued and how it is settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
       'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
       'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
    'A equity option simple strategy feature.'

   <strike> EquityStrike </strike> [1]
    'Defines whether it is a price or level at which the option has been, or will be, struck.'

   <spotPrice> NonNegativeDecimal </spotPrice> [0..1]
    'The price per share, index or basket observed on the trade or effective date.'

   <numberOfOptions> NonNegativeDecimal </numberOfOptions> [1]
    'The number of options comprised in the option transaction.'

   <equityPremium> EquityPremium </equityPremium> [1]
    'The equity option premium payable by the buyer to the seller.'

   <exchangeLookAlike> xsd:boolean </exchangeLookAlike> [0..1]
```

```
'For a share option transaction, a flag used to indicate whether the transaction is to be treated as an \'exchange look-alike\'. This
designation has significance for how share adjustments (arising from corporate actions) will be determined for the transaction. For an
\'exchange look-alike\' transaction the relevant share adjustments will follow that for a corresponding designated contract listed on the
related exchange (referred to as Options Exchange Adjustment (ISDA defined term), otherwise the share adjustments will be determined by the
calculation agent (referred to as Calculation Agent Adjustment (ISDA defined term)).'

<exchangeTradedContractNearest> xsd:boolean </exchangeTradedContractNearest> [0..1]
'For an index option transaction, a flag used in conjuction with Futures Price Valuation (ISDA defined term) to indicate whether the Nearest
Index Contract provision is applicable. The Nearest Index Contract provision is a rule for determining the Exchange-traded Contract (ISDA
defined term) without having to explicitly state the actual contract, delivery month and exchange on which it is traded.'

Start Group: IndexAnnexFallback.model [0..1]
   Start Choice [1]
      <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
      'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the transaction.
      This annex defines additional provisions which are applicable where an index is comprised of component securities that are traded on
      multiple exchanges.'

      <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
      'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the
      transaction.'

   End Choice
End Group: IndexAnnexFallback.model
<methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [0..1]
<localJurisdiction> Country </localJurisdiction> [0..1]
'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes, duties,
and similar charges imposed by the taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction is Not
Applicable.'

Start Choice [0..1]
   <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
   'The number of shares per option comprised in the option transaction supplement.'

   <multiplier> PositiveDecimal </multiplier> [1]
   'Specifies the contract multiplier that can be associated with an index option.'

End Choice
</equityOptionTransactionSupplement>
```

Diagram



Schema Component Representation

```
<xsd:element name="equityOptionTransactionSupplement" type=" EquityOptionTransactionSupplement " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: EquityExpiration.model

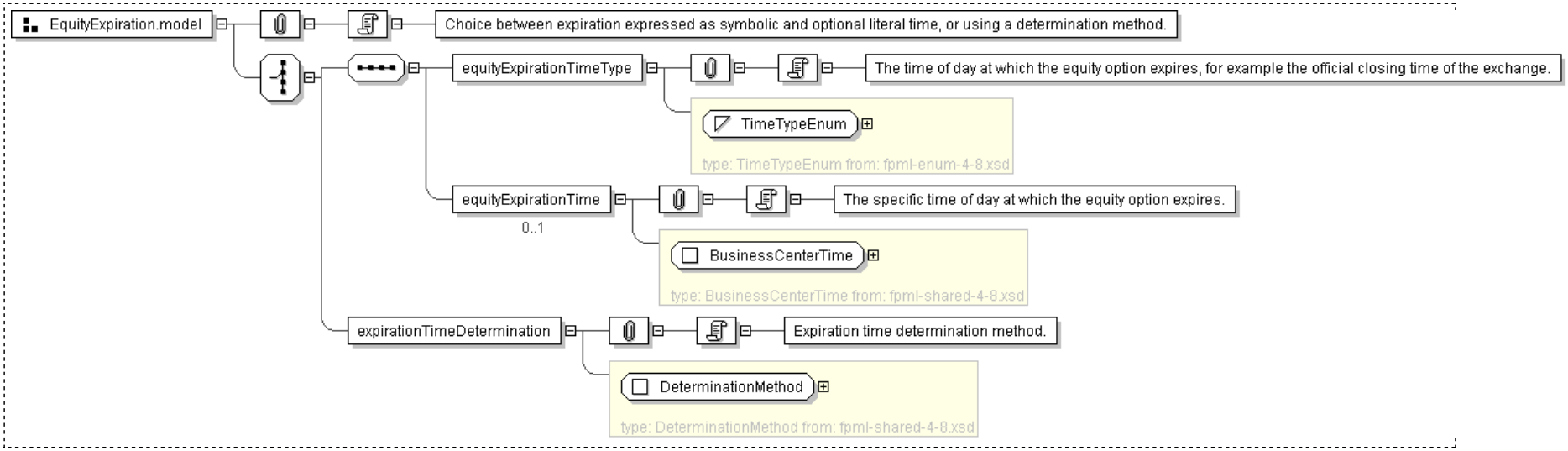
[Table of contents]

Name	EquityExpiration.model
Used by (from the same schema document)	Complex Type EquityAmericanExercise , Complex Type EquityBermudaExercise , Complex Type EquityEuropeanExercise
Documentation	Choice between expiration expressed as symbolic and optional literal time, or using a determination method.

XML Instance Representation

```
Start Choice [1]
   <equityExpirationTimeType> TimeTypeEnum </equityExpirationTimeType> [1]
   'The time of day at which the equity option expires, for example the official closing time of the exchange.'

   <equityExpirationTime> BusinessCenterTime </equityExpirationTime> [0..1]
   'The specific time of day at which the equity option expires.'

   <expirationTimeDetermination> DeterminationMethod </expirationTimeDetermination> [1]
   'Expiration time determination method.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="EquityExpiration.model">
   <xsd:choice>
      <xsd:sequence>
         <xsd:element name="equityExpirationTimeType" type=" TimeTypeEnum "/>
         <xsd:element name="equityExpirationTime" type=" BusinessCenterTime " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="expirationTimeDetermination" type=" DeterminationMethod "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BrokerEquityOption

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeShortFormBase (by extension) < BrokerEquityOption (by extension)
Sub-types:	None

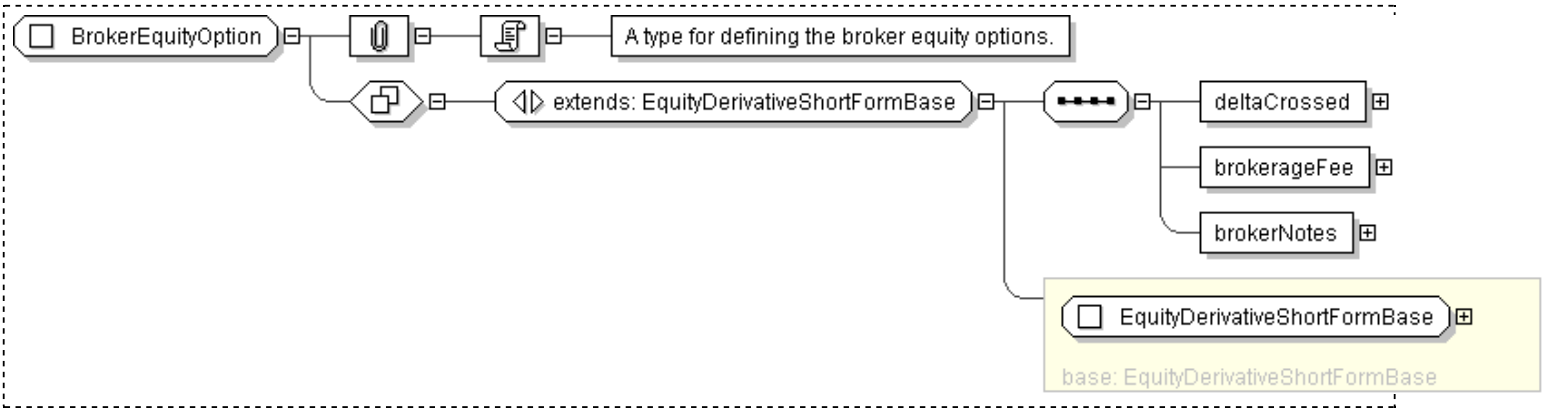
Name	BrokerEquityOption
Used by (from the same schema document)	Element brokerEquityOption
Abstract	no
Documentation	A type for defining the broker equity options.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization
   using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the
   case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the
   rights defined by this instrument and in return receives a payment for it. See 2000 ISDA
   definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued
   and how it is settled.'

   Start Group: Feature.model [0..1]
     <feature> OptionFeatures </feature> [0..1]
     'Asian, Barrier, Knock and Pass Through features.'

     <fxFeature> FxFeature </fxFeature> [0..1]
     'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <strike> EquityStrike </strike> [1]
   'Defines whether it is a price or level at which the option has been, or will be,
   struck.'
```

```
   <spotPrice> NonNegativeDecimal </spotPrice> [0..1]
   'The price per share, index or basket observed on the trade or effective date.'

   <numberOfOptions> NonNegativeDecimal </numberOfOptions> [1]
   'The number of options comprised in the option transaction.'

   <equityPremium> EquityPremium </equityPremium> [1]
   'The equity option premium payable by the buyer to the seller.'

   <deltaCrossed> xsd:boolean </deltaCrossed> [1]
   <brokerageFee> Money </brokerageFee> [1]
   <brokerNotes> xsd:string </brokerNotes> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BrokerEquityOption">
   <xsd:complexContent>
      <xsd:extension base=" EquityDerivativeShortFormBase ">
         <xsd:sequence>
            <xsd:element name="deltaCrossed" type=" xsd:boolean "/>
            <xsd:element name="brokerageFee" type=" Money "/>
            <xsd:element name="brokerNotes" type=" xsd:string "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityAmericanExercise

[Table of contents]

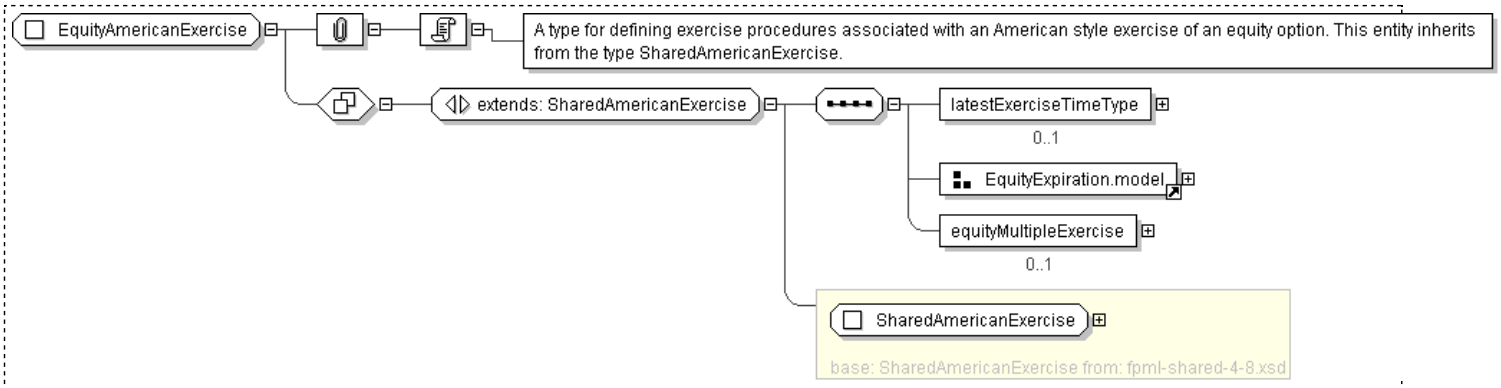
Super-types:	SharedAmericanExercise < EquityAmericanExercise (by extension)
Sub-types:	None

Name	EquityAmericanExercise
Used by (from the same schema document)	Complex Type EquityExerciseValuationSettlement
Abstract	no
Documentation	A type for defining exercise procedures associated with an American style exercise of an equity option. This entity inherits from the type SharedAmericanExercise.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
   'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the only
   day within the exercise period.'

   Start Choice [0..1]
'Choice between latest exercise time expressed as literal time, or using a determination method.'


      <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [1]
      'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration
      date) within the exercise period that notice can be given by the buyer to the seller or seller\'s agent. Notice
      of exercise given after this time will be deemed to have been given on the next exercise business day.'

      <latestExerciseTimeDetermination> DeterminationMethod </latestExerciseTimeDetermination> [1]
      'Latest exercise time determination method.'

   End Choice
   <latestExerciseTimeType> TimeTypeEnum </latestExerciseTimeType> [0..1]
   'The latest time of day at which the equity option can be exercised, for example the official closing time of the
   exchange.'

   Start Choice [1]
      <equityExpirationTimeType> TimeTypeEnum </equityExpirationTimeType> [1]
      'The time of day at which the equity option expires, for example the official closing time of the exchange.'

      <equityExpirationTime> BusinessCenterTime </equityExpirationTime> [0..1]
      'The specific time of day at which the equity option expires.'

      <expirationTimeDetermination> DeterminationMethod </expirationTimeDetermination> [1]
      'Expiration time determination method.'

   End Choice
   <equityMultipleExercise> EquityMultipleExercise </equityMultipleExercise> [0..1]
   'The presence of this element indicates that the option may be exercised on different days. It is not applicable
   to European options.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityAmericanExercise">
  <xsd:complexContent>
    <xsd:extension base=" SharedAmericanExercise ">
      <xsd:sequence>
        <xsd:element name="latestExerciseTimeType" type=" TimeTypeEnum " minOccurs="0"/>
```

```
            <xsd:group ref=" EquityExpiration.model "/>
            <xsd:element name="equityMultipleExercise" type=" EquityMultipleExercise " minOccurs="0"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityBermudaExercise

[Table of contents]

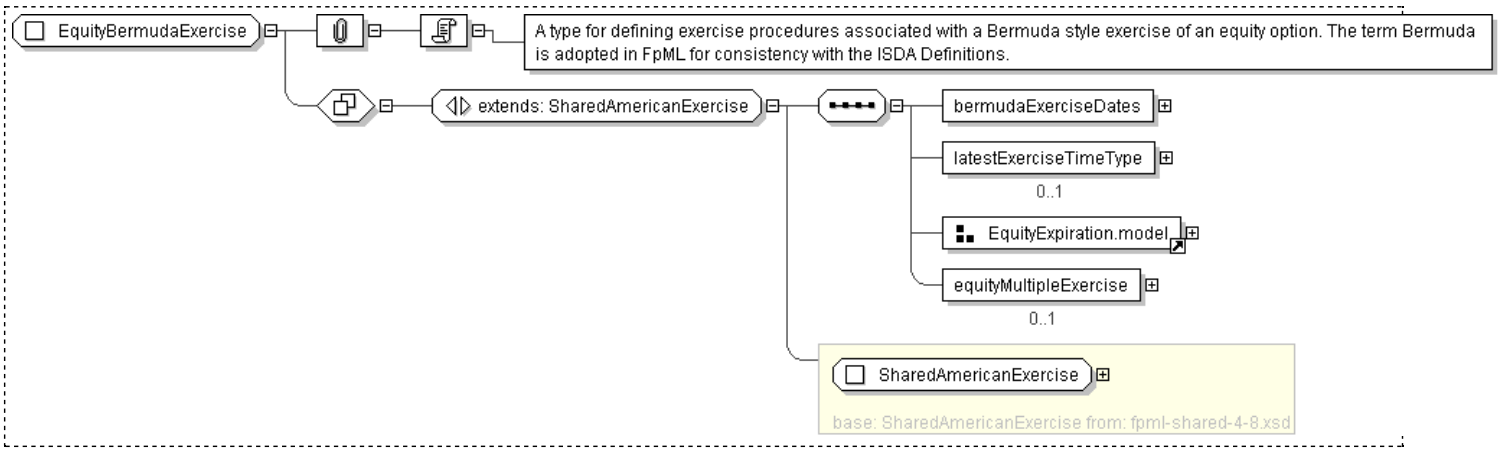
Super-types:	SharedAmericanExercise < EquityBermudaExercise (by extension)
Sub-types:	None

Name	EquityBermudaExercise
Used by (from the same schema document)	Complex Type EquityExerciseValuationSettlement
Abstract	no
Documentation	A type for defining exercise procedures associated with a Bermuda style exercise of an equity option. The term Bermuda is adopted in FpML for consistency with the ISDA Definitions.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
   'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the only
   day within the exercise period.'

   Start Choice [0..1]
'Choice between latest exercise time expressed as literal time, or using a determination method.'


      <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [1]
      'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration
      date) within the exercise period that notice can be given by the buyer to the seller or seller\'s agent. Notice
      of exercise given after this time will be deemed to have been given on the next exercise business day.'

      <latestExerciseTimeDetermination> DeterminationMethod </latestExerciseTimeDetermination> [1]
      'Latest exercise time determination method.'

   End Choice
   <bermudaExerciseDates> DateList </bermudaExerciseDates> [1]
   'List of Exercise Dates for a Bermuda option.'

   <latestExerciseTimeType> TimeTypeEnum </latestExerciseTimeType> [0..1]
   'The latest time of day at which the equity option can be exercised, for example the official closing time of the
   exchange.'

   Start Choice [1]
      <equityExpirationTimeType> TimeTypeEnum </equityExpirationTimeType> [1]
      'The time of day at which the equity option expires, for example the official closing time of the exchange.'

      <equityExpirationTime> BusinessCenterTime </equityExpirationTime> [0..1]
      'The specific time of day at which the equity option expires.'

      <expirationTimeDetermination> DeterminationMethod </expirationTimeDetermination> [1]
      'Expiration time determination method.'

   End Choice
   <equityMultipleExercise> EquityMultipleExercise </equityMultipleExercise> [0..1]
   'The presence of this element indicates that the option may be exercised on different days. It is not applicable
   to European options.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityBermudaExercise">
   <xsd:complexContent>
      <xsd:extension base=" SharedAmericanExercise ">
         <xsd:sequence>
            <xsd:element name="bermudaExerciseDates" type=" DateList "/>
            <xsd:element name="latestExerciseTimeType" type=" TimeTypeEnum " minOccurs="0"/>
            <xsd:group ref=" EquityExpiration.model "/>
            <xsd:element name="equityMultipleExercise" type=" EquityMultipleExercise " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityDerivativeBase

[Table of contents]

Super-types: Product < EquityDerivativeBase (by extension)

Sub-types:

- EquityDerivativeLongFormBase (by extension)
  - EquityForward (by extension)
  - EquityOption (by extension)
- EquityDerivativeShortFormBase (by extension)
  - BrokerEquityOption (by extension)
  - EquityOptionTransactionSupplement (by extension)

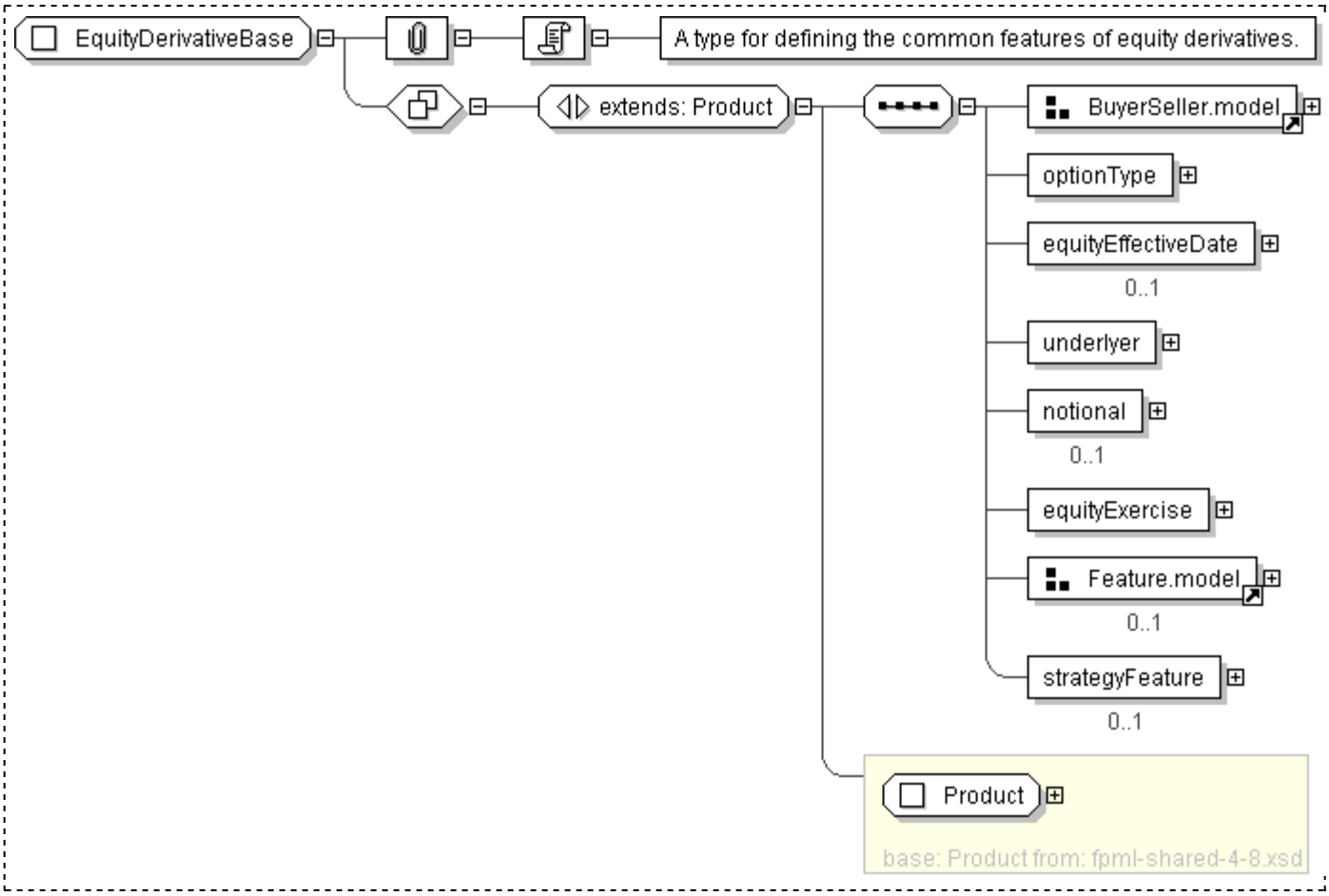
Name	EquityDerivativeBase
Abstract	yes
Documentation	A type for defining the common features of equity derivatives.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for
   this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this
   instrument and receives the rights defined by it. See 2000 ISDA definitions
   Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that
   grants the rights defined by this instrument and in return receives a payment
   for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is
   the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and
   consists in either equity, index or convertible bond component, or a
   combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is
   valued and how it is settled.'
```

```
Start Group: Feature.model [0..1]
   <feature> OptionFeatures </feature> [0..1]
   'Asian, Barrier, Knock and Pass Through features.'

   <fxFeature> FxFeature </fxFeature> [0..1]
   'Quanto, Composite, or Cross Currency FX features.'

End Group: Feature.model
<strategyFeature> StrategyFeature </strategyFeature> [0..1]
'A equity option simple strategy feature.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityDerivativeBase" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:group ref=" BuyerSeller.model "/>
            <xsd:element name="optionType" type=" OptionTypeEnum "/>
            <xsd:element name="equityEffectiveDate" type=" xsd:date " minOccurs="0"/>
            <xsd:element name="underlyer" type=" Underlyer "/>
            <xsd:element name="notional" type=" Money " minOccurs="0"/>
            <xsd:element name="equityExercise" type=" EquityExerciseValuationSettlement
            "/>
            <xsd:group ref=" Feature.model " minOccurs="0"/>
            <xsd:element name="strategyFeature" type=" StrategyFeature " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityDerivativeLongFormBase

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeLongFormBase (by extension)
Sub-types:	• EquityForward (by extension) • EquityOption (by extension)

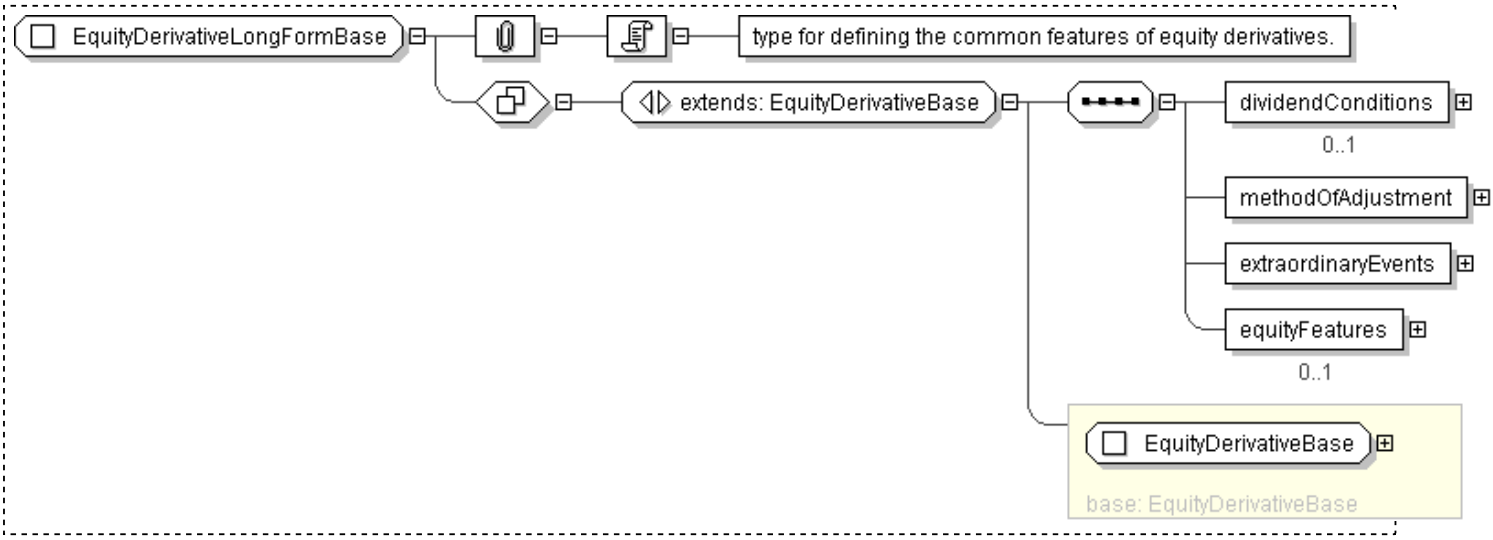
Name	EquityDerivativeLongFormBase
Abstract	yes
Documentation	type for defining the common features of equity derivatives.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization
   using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the
   case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the
   rights defined by this instrument and in return receives a payment for it. See 2000 ISDA
   definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued
   and how it is settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
      'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
      'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <dividendConditions> DividendConditions </dividendConditions> [0..1]
   <methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [1]
   'Defines how adjustments will be made to the contract should one or more of the
```

```
   extraordinary events occur.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [1]
   'Where the underlying is shares, specifies events affecting the issuer of those shares
   that may require the terms of the transaction to be adjusted.'

   <equityFeatures> OptionFeatures </equityFeatures> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Use the
   \"feature\" element for option features such as asian, barrier, knock.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityDerivativeLongFormBase" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" EquityDerivativeBase ">
         <xsd:sequence>
            <xsd:element name="dividendConditions" type=" DividendConditions " minOccurs="0"/>
            <xsd:element name="methodOfAdjustment" type=" MethodOfAdjustmentEnum "/>
            <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents "/>
            <xsd:element name="equityFeatures" type=" OptionFeatures " minOccurs="0"
            deprecated="true" deprecatedReason="Option Features content is accessible in the
            complex type EquityDerivativeBase through the model group Feature.model"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityDerivativeShortFormBase

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeShortFormBase (by extension)
Sub-types:	• BrokerEquityOption (by extension) • EquityOptionTransactionSupplement (by extension)

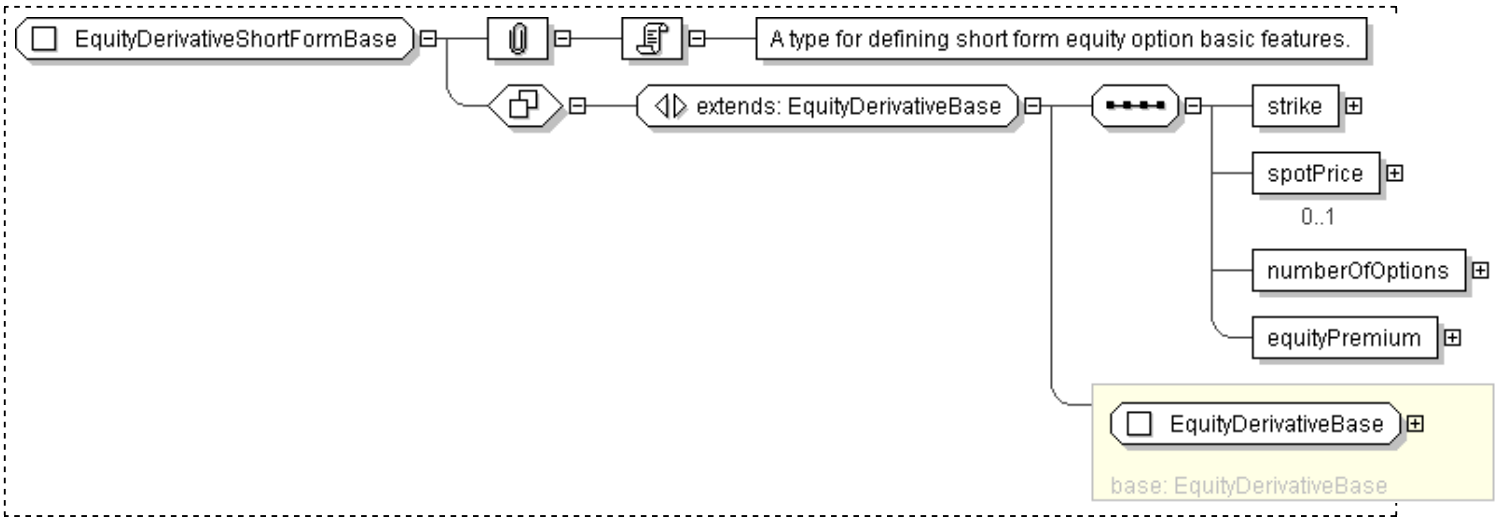
Name	EquityDerivativeShortFormBase
Abstract	yes
Documentation	A type for defining short form equity option basic features.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization
   using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the
   case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants
   the rights defined by this instrument and in return receives a payment for it. See
   2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued
   and how it is settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
      'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
      'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <strike> EquityStrike </strike> [1]
```

```
'Defines whether it is a price or level at which the option has been, or will be, struck.'

<spotPrice> NonNegativeDecimal </spotPrice> [0..1]
'The price per share, index or basket observed on the trade or effective date.'

<numberOfOptions> NonNegativeDecimal </numberOfOptions> [1]
'The number of options comprised in the option transaction.'

<equityPremium> EquityPremium </equityPremium> [1]
'The equity option premium payable by the buyer to the seller.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityDerivativeShortFormBase" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" EquityDerivativeBase ">
      <xsd:sequence>
        <xsd:element name="strike" type=" EquityStrike "/>
        <xsd:element name="spotPrice" type=" NonNegativeDecimal " minOccurs="0"/>
        <xsd:element name="numberOfOptions" type=" NonNegativeDecimal "/>
        <xsd:element name="equityPremium" type=" EquityPremium "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityEuropeanExercise

[Table of contents]

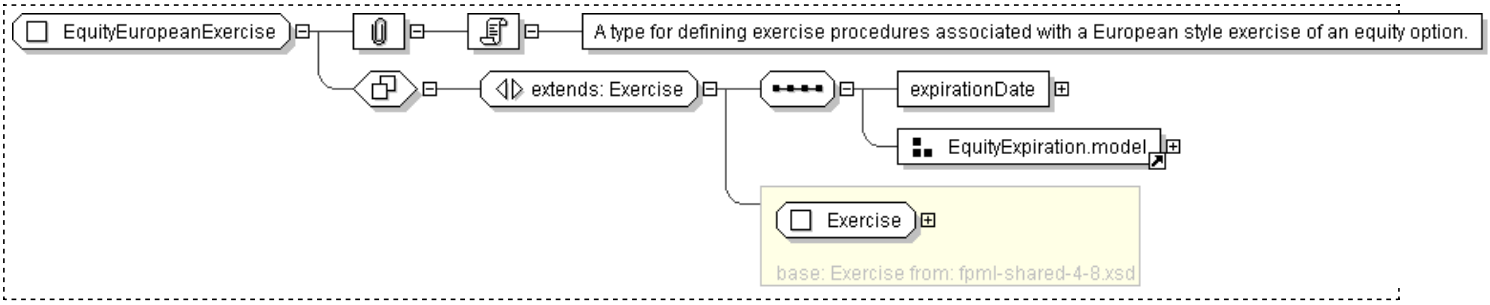
Super-types:	Exercise < EquityEuropeanExercise (by extension)
Sub-types:	None

Name	EquityEuropeanExercise
Used by (from the same schema document)	Complex Type EquityExerciseValuationSettlement
Abstract	no
Documentation	A type for defining exercise procedures associated with a European style exercise of an equity option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it
   is the only day within the exercise period.'

   Start Choice [1]
      <equityExpirationTimeType> TimeTypeEnum </equityExpirationTimeType> [1]
      'The time of day at which the equity option expires, for example the official closing time of the
      exchange.'

      <equityExpirationTime> BusinessCenterTime </equityExpirationTime> [0..1]
      'The specific time of day at which the equity option expires.'

      <expirationTimeDetermination> DeterminationMethod </expirationTimeDetermination> [1]
      'Expiration time determination method.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityEuropeanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
        <xsd:group ref=" EquityExpiration.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityExerciseValuationSettlement

[Table of contents]

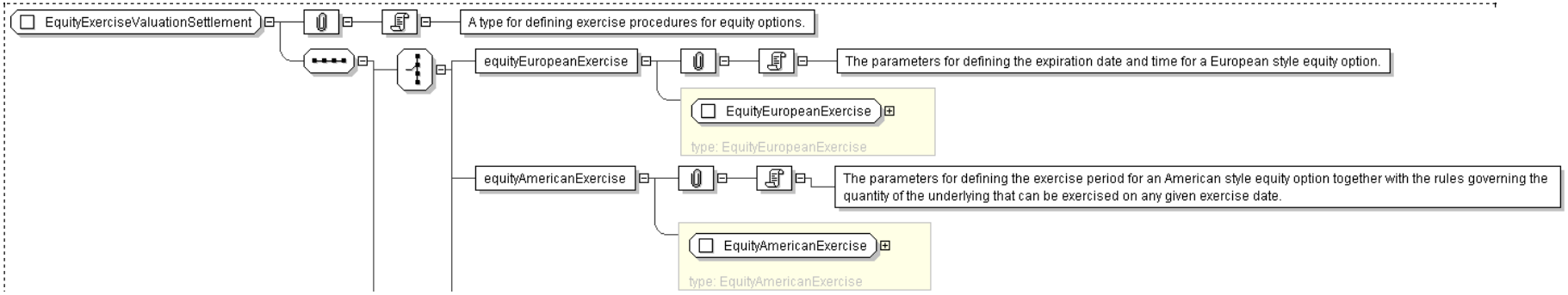
Super-types:	None
Sub-types:	None

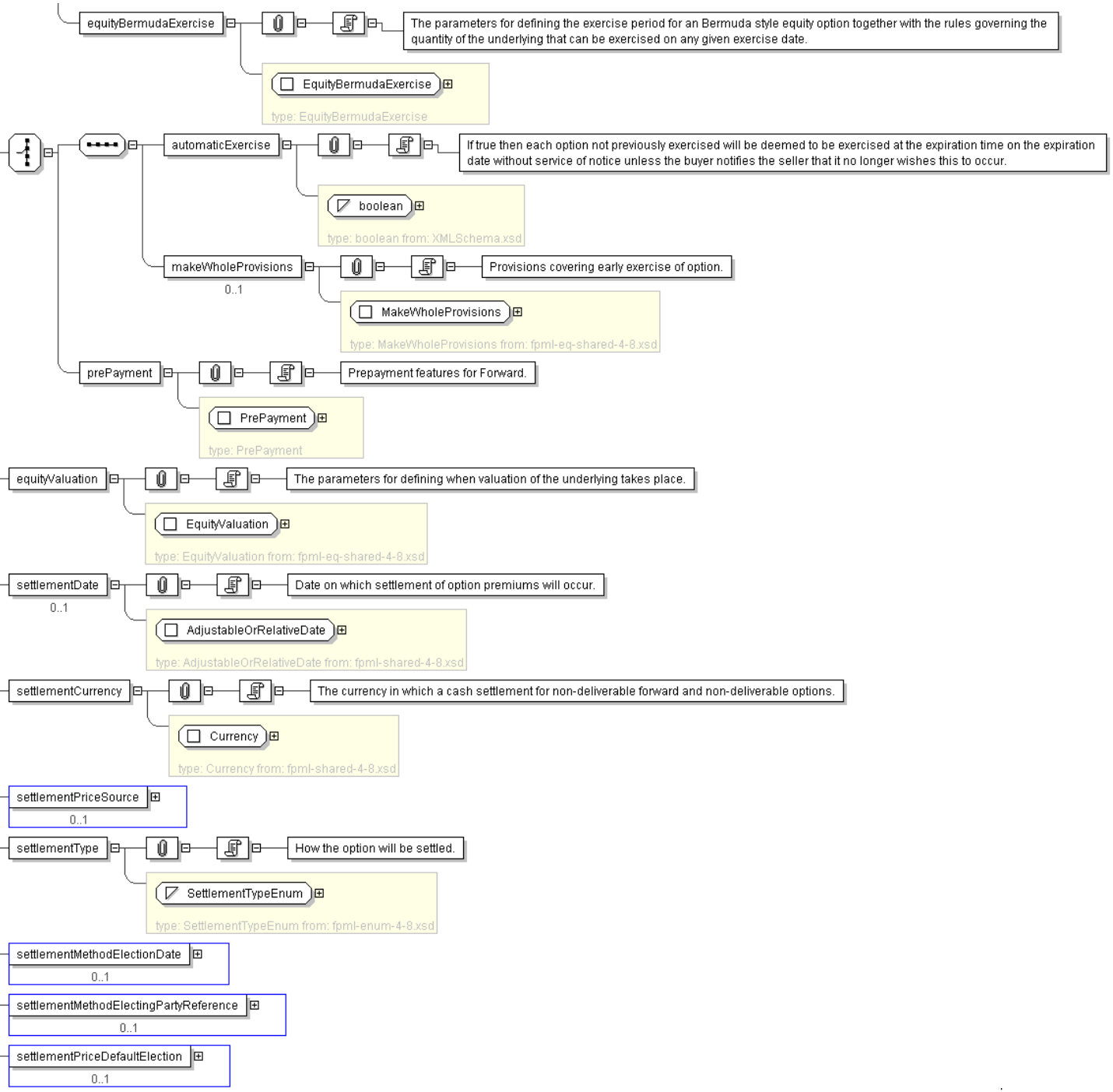
Name	EquityExerciseValuationSettlement
Used by (from the same schema document)	Complex Type EquityDerivativeBase
Abstract	no
Documentation	A type for defining exercise procedures for equity options.

XML Instance Representation

```
<...>
  Start Choice [1]
    <equityEuropeanExercise> EquityEuropeanExercise </equityEuropeanExercise> [1]
    'The parameters for defining the expiration date and time for a European style equity option.'

    <equityAmericanExercise> EquityAmericanExercise </equityAmericanExercise> [1]
    'The parameters for defining the exercise period for an American style equity option together with the rules governing the quantity of the underlying that can be
    exercised on any given exercise date.'

    <equityBermudaExercise> EquityBermudaExercise </equityBermudaExercise> [1]
    'The parameters for defining the exercise period for an Bermuda style equity option together with the rules governing the quantity of the underlying that can be
    exercised on any given exercise date.'

  End Choice
  Start Choice [1]
    <automaticExercise> xsd:boolean </automaticExercise> [1]
    'If true then each option not previously exercised will be deemed to be exercised at the expiration time on the expiration date without service of notice unless
    the buyer notifies the seller that it no longer wishes this to occur.'

    <makeWholeProvisions> MakeWholeProvisions </makeWholeProvisions> [0..1]
    'Provisions covering early exercise of option.'

    <prePayment> PrePayment </prePayment> [1]
    'Prepayment features for Forward.'

  End Choice
  <equityValuation> EquityValuation </equityValuation> [1]
  'The parameters for defining when valuation of the underlying takes place.'

  <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
  'Date on which settlement of option premiums will occur.'

  <settlementCurrency> Currency </settlementCurrency> [1]
  'The currency in which a cash settlement for non-deliverable forward and non-deliverable options.'

  <settlementPriceSource> SettlementPriceSource </settlementPriceSource> [0..1]
  <settlementType> SettlementTypeEnum </settlementType> [1]
  'How the option will be settled.'

  <settlementMethodElectionDate> AdjustableOrRelativeDate </settlementMethodElectionDate> [0..1]
  <settlementMethodElectingPartyReference> PartyReference </settlementMethodElectingPartyReference> [0..1]
  <settlementPriceDefaultElection> SettlementPriceDefaultElection </settlementPriceDefaultElection> [0..1]
</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="EquityExerciseValuationSettlement">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="equityEuropeanExercise" type=" EquityEuropeanExercise "/>
      <xsd:element name="equityAmericanExercise" type=" EquityAmericanExercise "/>
```

```
        <xsd:element name="equityBermudaExercise" type=" EquityBermudaExercise "/>
      </xsd:choice>
      <xsd:choice>
        <xsd:sequence>
          <xsd:element name="automaticExercise" type=" xsd:boolean "/>
          <xsd:element name="makeWholeProvisions" type=" MakeWholeProvisions " minOccurs="0"/>
        </xsd:sequence>
        <xsd:element name="prePayment" type=" PrePayment "/>
      </xsd:choice>
      <xsd:element name="equityValuation" type=" EquityValuation "/>
      <xsd:element name="settlementDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      <xsd:element name="settlementCurrency" type=" Currency "/>
      <xsd:element name="settlementPriceSource" type=" SettlementPriceSource " minOccurs="0"/>
      <xsd:element name="settlementType" type=" SettlementTypeEnum "/>
      <xsd:element name="settlementMethodElectionDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      <xsd:element name="settlementMethodElectingPartyReference" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="settlementPriceDefaultElection" type=" SettlementPriceDefaultElection " minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityForward

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeLongFormBase (by extension) < EquityForward (by extension)
Sub-types:	None

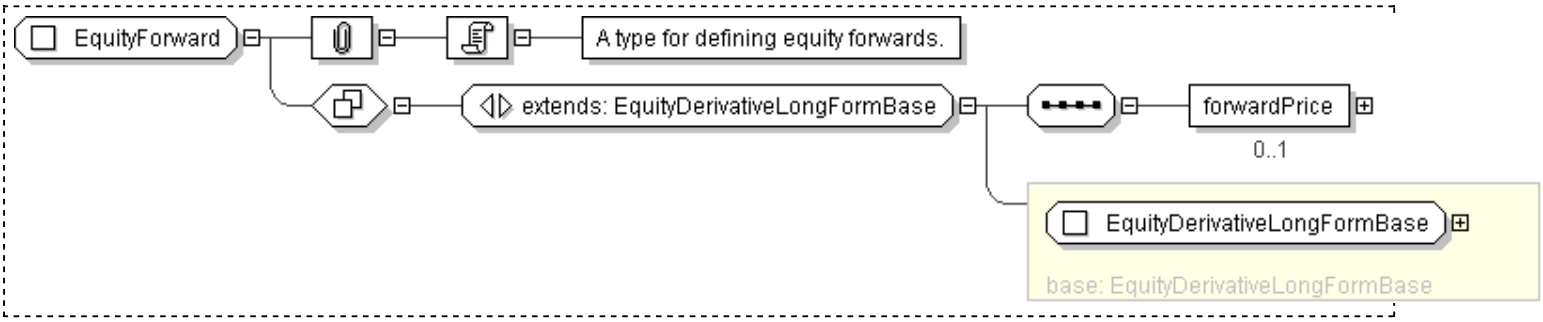
Name	EquityForward
Used by (from the same schema document)	Element equityForward
Abstract	no
Documentation	A type for defining equity forwards.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and
   receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the
   case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants
   the rights defined by this instrument and in return receives a payment for it. See
   2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued
   and how it is settled.'

   Start Group: Feature.model [0..1]
     <feature> OptionFeatures </feature> [0..1]
     'Asian, Barrier, Knock and Pass Through features.'

     <fxFeature> FxFeature </fxFeature> [0..1]
     'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'
```

```
<dividendConditions> DividendConditions </dividendConditions> [0..1]
<methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [1]
'Defines how adjustments will be made to the contract should one or more of the
extraordinary events occur.'

<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [1]
'Where the underlying is shares, specifies events affecting the issuer of those
shares that may require the terms of the transaction to be adjusted.'

<equityFeatures> OptionFeatures </equityFeatures> [0..1]
'DEPRECATED This element will be removed in the next FpML major version. Use the
\"feature\" element for option features such as asian, barrier, knock.'

<forwardPrice> Money </forwardPrice> [0..1]
'The forward price per share, index or basket.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityForward">
  <xsd:complexContent>
    <xsd:extension base=" EquityDerivativeLongFormBase ">
      <xsd:sequence>
        <xsd:element name="forwardPrice" type=" Money " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityMultipleExercise

[Table of contents]

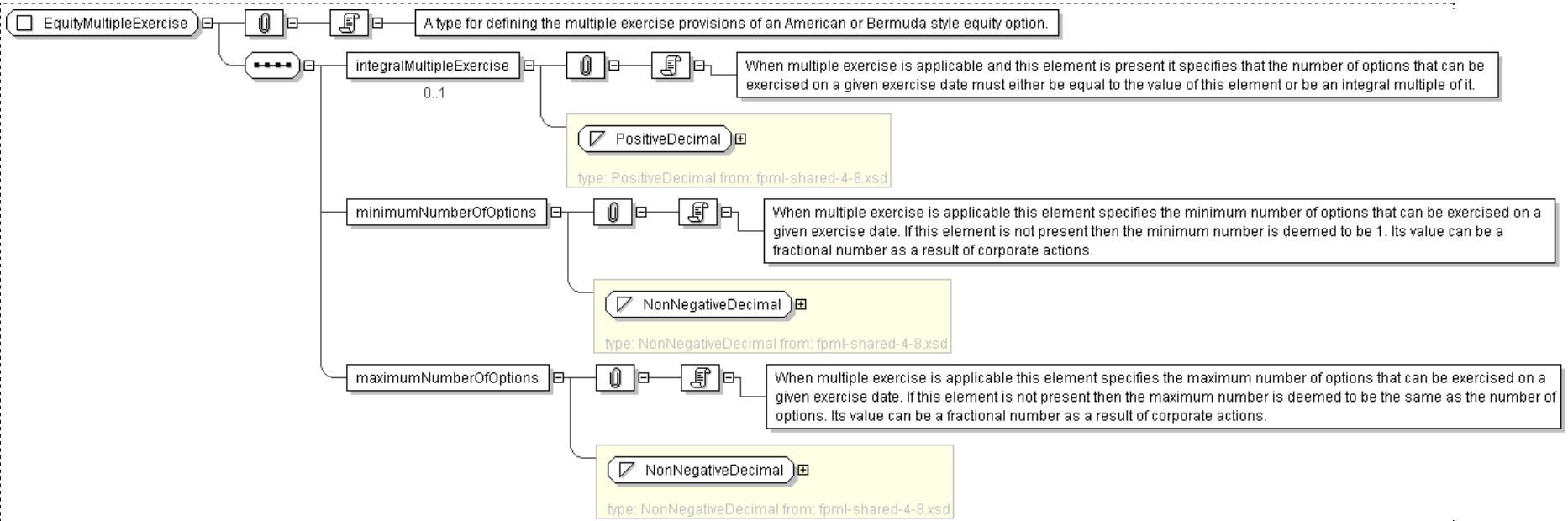
Super-types:	None
Sub-types:	None

Name	EquityMultipleExercise
Used by (from the same schema document)	Complex Type EquityAmericanExercise , Complex Type EquityBermudaExercise
Abstract	no
Documentation	A type for defining the multiple exercise provisions of an American or Bermuda style equity option.

XML Instance Representation

```
<...>
   <integralMultipleExercise> PositiveDecimal </integralMultipleExercise> [0..1]
   'When multiple exercise is applicable and this element is present it specifies that the number of options that can be exercised on a given exercise
   date must either be equal to the value of this element or be an integral multiple of it.'

   <minimumNumberOfOptions> NonNegativeDecimal </minimumNumberOfOptions> [1]
   'When multiple exercise is applicable this element specifies the minimum number of options that can be exercised on a given exercise date. If this
   element is not present then the minimum number is deemed to be 1. Its value can be a fractional number as a result of corporate actions.'

   <maximumNumberOfOptions> NonNegativeDecimal </maximumNumberOfOptions> [1]
   'When multiple exercise is applicable this element specifies the maximum number of options that can be exercised on a given exercise date. If this
   element is not present then the maximum number is deemed to be the same as the number of options. Its value can be a fractional number as a result
   of corporate actions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityMultipleExercise">
   <xsd:sequence>
```

```
      <xsd:element name="integralMultipleExercise" type=" PositiveDecimal " minOccurs="0"/>
      <xsd:element name="minimumNumberOfOptions" type=" NonNegativeDecimal "/>
      <xsd:element name="maximumNumberOfOptions" type=" NonNegativeDecimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityOption

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeLongFormBase (by extension) < EquityOption (by extension)
Sub-types:	None

Name	EquityOption
Used by (from the same schema document)	Element equityOption
Abstract	no
Documentation	A type for defining equity options.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for this
   element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument
   and receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b).
   In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants
   the rights defined by this instrument and in return receives a payment for it. See
   2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating
   rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is
   valued and how it is settled.'

   Start Group: Feature.model [0..1]
      <feature> OptionFeatures </feature> [0..1]
      'Asian, Barrier, Knock and Pass Through features.'

      <fxFeature> FxFeature </fxFeature> [0..1]
      'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'
```

<dividendConditions> DividendConditions </dividendConditions> [0..1]
<methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [1]
'Defines how adjustments will be made to the contract should one or more of the extraordinary events occur.'

<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [1]
'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.'

<equityFeatures> OptionFeatures </equityFeatures> [0..1]
'DEPRECATED This element will be removed in the next FpML major version. Use the \"feature\" element for option features such as asian, barrier, knock.'

<strike> EquityStrike </strike> [0..1]
'Defines whether it is a price or level at which the option has been, or will be, struck.'

<spotPrice> NonNegativeDecimal </spotPrice> [0..1]
'The price per share, index or basket observed on the trade or effective date.'

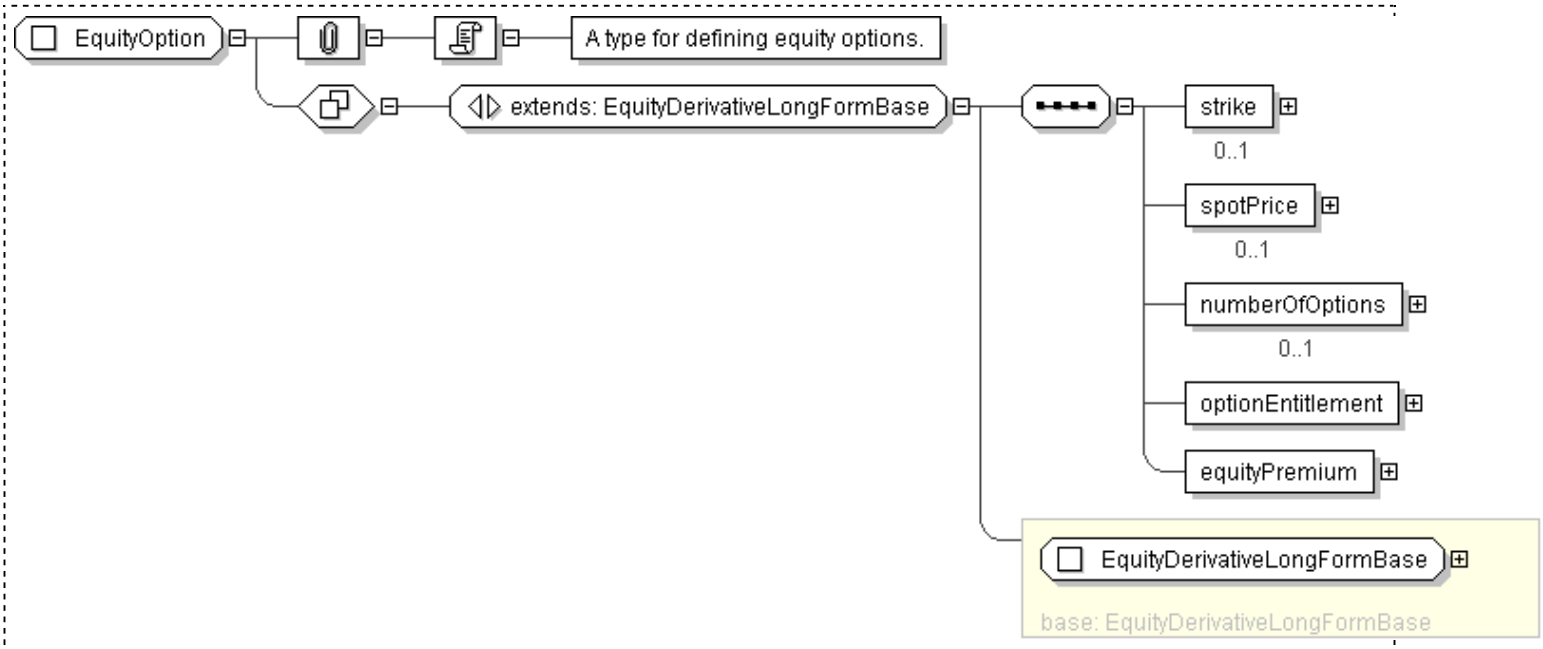
<numberOfOptions> NonNegativeDecimal </numberOfOptions> [0..1]
'The number of options comprised in the option transaction.'

<optionEntitlement> PositiveDecimal </optionEntitlement> [1]
'The number of shares per option comprised in the option transaction.'

<equityPremium> EquityPremium </equityPremium> [1]
'The equity option premium payable by the buyer to the seller.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityOption">
   <xsd:complexContent>
      <xsd:extension base=" EquityDerivativeLongFormBase ">
         <xsd:sequence>
            <xsd:element name="strike" type=" EquityStrike " minOccurs="0"/>
            <xsd:element name="spotPrice" type=" NonNegativeDecimal " minOccurs="0"/>
            <xsd:element name="numberOfOptions" type=" NonNegativeDecimal " minOccurs="0"/>
            <xsd:element name="optionEntitlement" type=" PositiveDecimal "/>
            <xsd:element name="equityPremium" type=" EquityPremium "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityOptionTermination

[Table of contents]

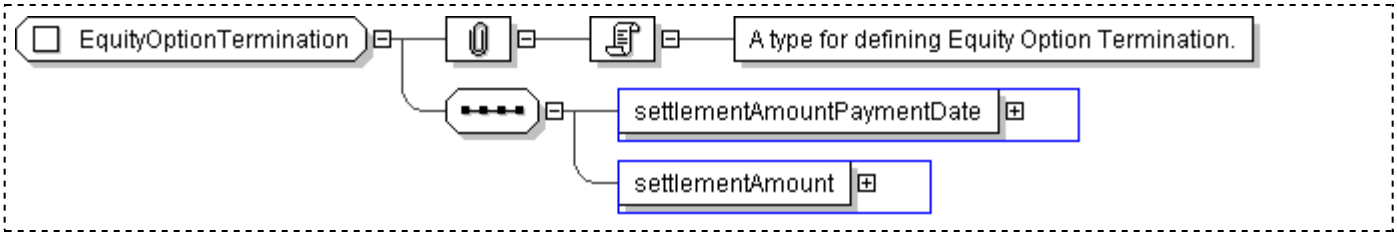
Super-types:	None
Sub-types:	None

Name	EquityOptionTermination
Abstract	no
Documentation	A type for defining Equity Option Termination.

XML Instance Representation

```
<...>
   <settlementAmountPaymentDate> AdjustableDate </settlementAmountPaymentDate> [1]
   <settlementAmount> Money </settlementAmount> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityOptionTermination">
   <xsd:sequence>
      <xsd:element name="settlementAmountPaymentDate" type=" AdjustableDate "/>
      <xsd:element name="settlementAmount" type=" Money "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityOptionTransactionSupplement

[Table of contents]

Super-types:	Product < EquityDerivativeBase (by extension) < EquityDerivativeShortFormBase (by extension) < EquityOptionTransactionSupplement (by extension)
Sub-types:	None

Name	EquityOptionTransactionSupplement
Used by (from the same schema document)	Element equityOptionTransactionSupplement
Abstract	no
Documentation	A type for defining equity option transaction supplements.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated
   with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate
   payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In
   the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction.'

   <equityEffectiveDate> xsd:date </equityEffectiveDate> [0..1]
   'Effective date for a forward starting option.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component, which can be either one or many and consists in either equity, index
   or convertible bond component, or a combination of these.'

   <notional> Money </notional> [0..1]
   'The notional amount.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued and how it is
   settled.'

   Start Group: Feature.model [0..1]
     <feature> OptionFeatures </feature> [0..1]
     'Asian, Barrier, Knock and Pass Through features.'

     <fxFeature> FxFeature </fxFeature> [0..1]
     'Quanto, Composite, or Cross Currency FX features.'

   End Group: Feature.model
   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
   'A equity option simple strategy feature.'

   <strike> EquityStrike </strike> [1]
   'Defines whether it is a price or level at which the option has been, or will be, struck.'

   <spotPrice> NonNegativeDecimal </spotPrice> [0..1]
   'The price per share, index or basket observed on the trade or effective date.'

   <numberOfOptions> NonNegativeDecimal </numberOfOptions> [1]
   'The number of options comprised in the option transaction.'

   <equityPremium> EquityPremium </equityPremium> [1]
   'The equity option premium payable by the buyer to the seller.'

   <exchangeLookAlike> xsd:boolean </exchangeLookAlike> [0..1]
   'For a share option transaction, a flag used to indicate whether the transaction is to be treated as an
   \'exchange look-alike\'. This designation has significance for how share adjustments (arising from
   corporate actions) will be determined for the transaction. For an \'exchange look-alike\' transaction the
   relevant share adjustments will follow that for a corresponding designated contract listed on the related
```

exchange (referred to as Options Exchange Adjustment (ISDA defined term), otherwise the share adjustments will be determined by the calculation agent (referred to as Calculation Agent Adjustment (ISDA defined term)).'

<exchangeTradedContractNearest> xsd:boolean **</exchangeTradedContractNearest>** [0..1]

'For an index option transaction, a flag used in conjuction with Futures Price Valuation (ISDA defined term) to indicate whether the Nearest Index Contract provision is applicable. The Nearest Index Contract provision is a rule for determining the Exchange-traded Contract (ISDA defined term) without having to explicitly state the actual contract, delivery month and exchange on which it is traded.'

Start Group: IndexAnnexFallback.model [0..1]
Start Choice [1]

<multipleExchangeIndexAnnexFallback> xsd:boolean **</multipleExchangeIndexAnnexFallback>** [1]

'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the transaction. This annex defines additional provisions which are applicable where an index is comprised of component securities that are traded on multiple exchanges.'

<componentSecurityIndexAnnexFallback> xsd:boolean **</componentSecurityIndexAnnexFallback>** [1]

'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the transaction.'

End Choice
End Group: IndexAnnexFallback.model

<methodOfAdjustment> MethodOfAdjustmentEnum **</methodOfAdjustment>** [0..1]

<localJurisdiction> Country **</localJurisdiction>** [0..1]

'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes, duties, and similar charges imposed by the taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction is Not Applicable.'

Start Choice [0..1]

<optionEntitlement> PositiveDecimal **</optionEntitlement>** [1]

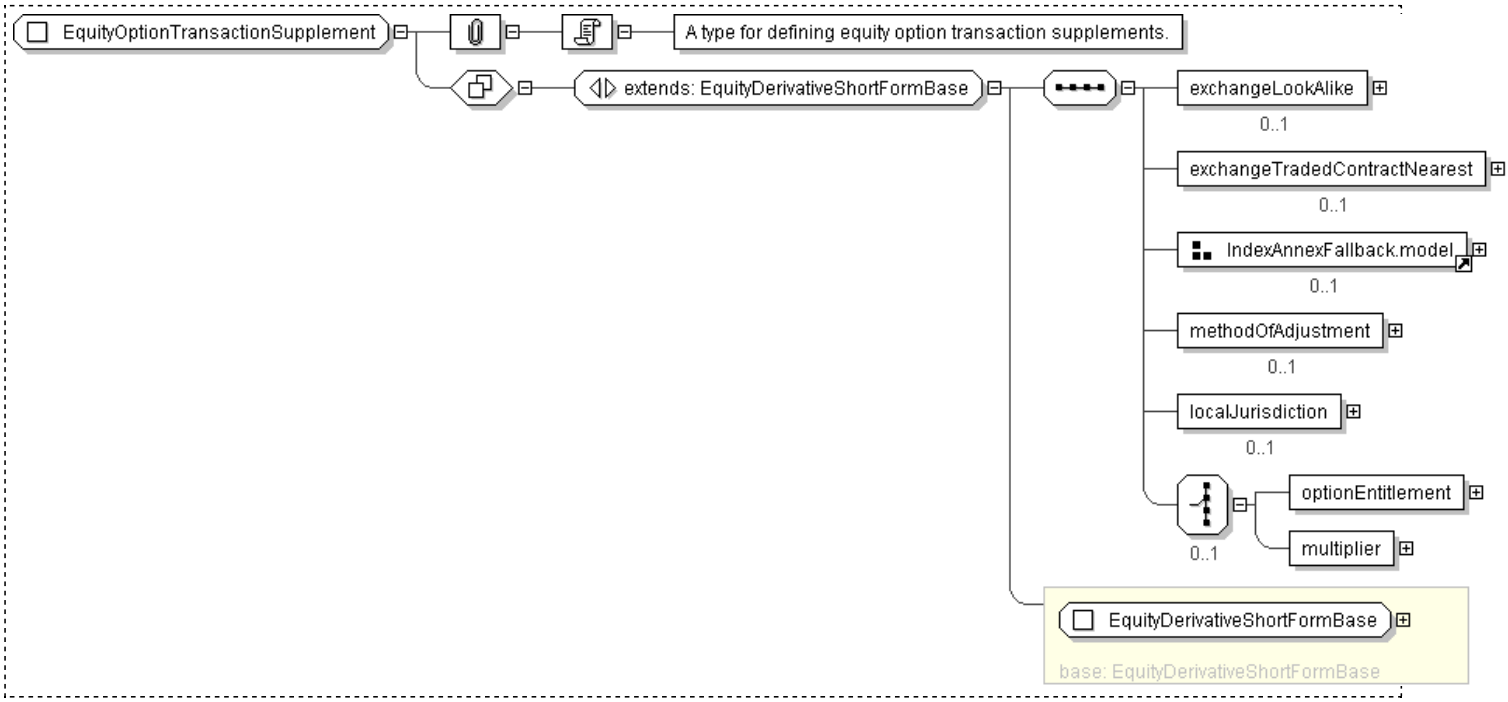
'The number of shares per option comprised in the option transaction supplement.'

<multiplier> PositiveDecimal **</multiplier>** [1]

'Specifies the contract multiplier that can be associated with an index option.'

End Choice
</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityOptionTransactionSupplement">
  <xsd:complexContent>
    <xsd:extension base=" EquityDerivativeShortFormBase ">
      <xsd:sequence>
        <xsd:element name="exchangeLookAlike" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="exchangeTradedContractNearest" type=" xsd:boolean " minOccurs="0"/>
        <xsd:group ref=" IndexAnnexFallback.model " minOccurs="0"/>
        <xsd:element name="methodOfAdjustment" type=" MethodOfAdjustmentEnum " minOccurs="0"/>
        <xsd:element name="localJurisdiction" type=" Country " minOccurs="0"/>
        <xsd:choice minOccurs="0">
          <xsd:element name="optionEntitlement" type=" PositiveDecimal "/>
          <xsd:element name="multiplier" type=" PositiveDecimal "/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrePayment

[Table of contents]

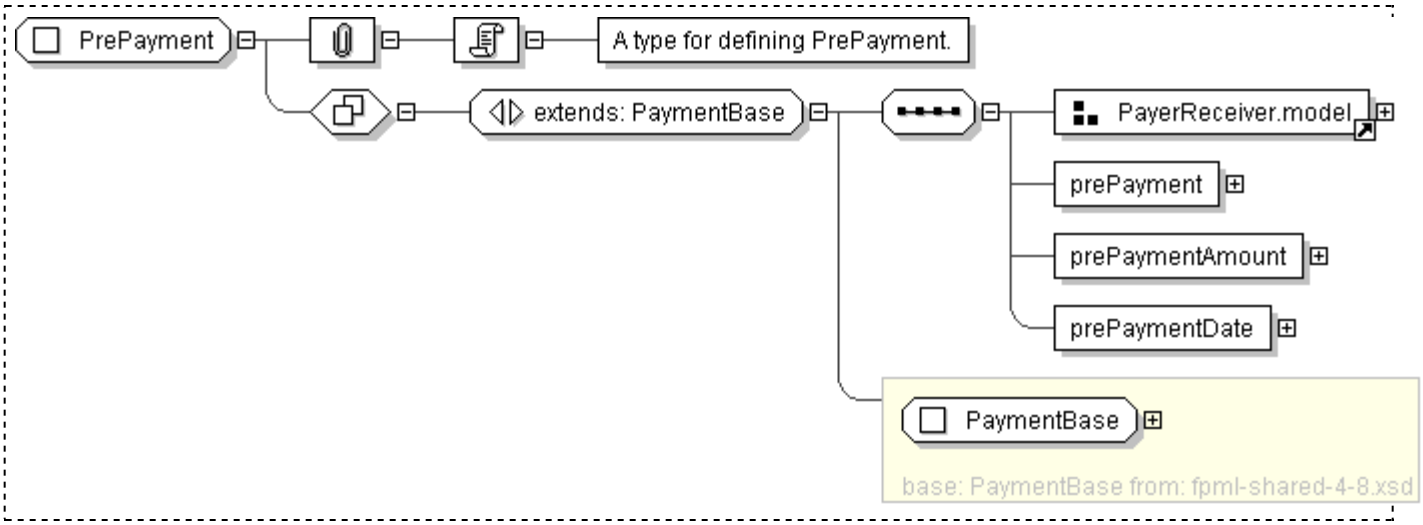
Super-types:	PaymentBase < PrePayment (by extension)
Sub-types:	None

Name	PrePayment
Used by (from the same schema document)	Complex Type EquityExerciseValuationSettlement
Abstract	no
Documentation	A type for defining PrePayment.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <prePayment> xsd:boolean </prePayment> [1]
   <prePaymentAmount> Money </prePaymentAmount> [1]
   <prePaymentDate> AdjustableDate </prePaymentDate> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="prePayment" type=" xsd:boolean "/>
            <xsd:element name="                    " type=" _____ "/>
```

```
                                prePaymentAmount          Money
            <xsd:element name="prePaymentDate" type=" AdjustableDate "/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: interestLeg
  - Element: returnLeg
  - Element: returnSwap
  - Element: returnSwapLeg
  - Element: varianceLeg
- Global Definitions
  - Complex Type: AdditionalDisruptionEvents
  - Complex Type: AdditionalPaymentAmount
  - Complex Type: AdjustableDateOrRelativeDateSequence
  - Complex Type: BoundedCorrelation
  - Complex Type: BoundedVariance
  - Complex Type: CalculatedAmount
  - Complex Type: CalculationFromObservation
  - Complex Type: Compounding
  - Complex Type: CompoundingRate
  - Complex Type: Correlation
  - Complex Type: DeprecatedVariance
  - Complex Type: DeprecatedVarianceAmount
  - Complex Type: DeprecatedVarianceLeg
  - Complex Type: DirectionalLeg
  - Complex Type: DirectionalLegUnderlyer
  - Complex Type: DirectionalLegUnderlyerValuation
  - Complex Type: DividendAdjustment
  - Complex Type: DividendConditions
  - Complex Type: DividendPaymentDate
  - Complex Type: DividendPeriod
  - Complex Type: DividendPeriodDividend
  - Complex Type: EquityCorporateEvents
  - Complex Type: EquityPremium
  - Complex Type: EquityStrike
  - Complex Type: EquityValuation
  - Complex Type: ExtraordinaryEvents
  - Complex Type: FloatingRateCalculationReference
  - Complex Type: IndexAdjustmentEvents
  - Complex Type: InterestCalculation
  - Complex Type: InterestLeg
  - Complex Type: InterestLegCalculationPeriodDates
  - Complex Type: InterestLegCalculationPeriodDatesReference
  - Complex Type: InterestLegResetDates
  - Complex Type: LegAmount
  - Complex Type: LegId
  - Complex Type: LegIdentifier
  - Complex Type: MakeWholeProvisions
  - Complex Type: NettedSwapBase
  - Complex Type: OptionFeatures
  - Complex Type: PrincipalExchangeAmount
  - Complex Type: PrincipalExchangeDescriptions
  - Complex Type: PrincipalExchangeFeatures
  - Complex Type: Representations
  - Complex Type: Return
  - Complex Type: ReturnLeg
  - Complex Type: ReturnLegValuation

- Complex Type: ReturnLegValuationPrice
- Complex Type: ReturnSwap
- Complex Type: ReturnSwapAdditionalPayment
- Complex Type: ReturnSwapAmount
- Complex Type: ReturnSwapBase
- Complex Type: ReturnSwapEarlyTermination
- Complex Type: ReturnSwapLeg
- Complex Type: ReturnSwapLegUnderlyer
- Complex Type: ReturnSwapNotional
- Complex Type: ReturnSwapPaymentDates
- Complex Type: StartingDate
- Complex Type: StubCalculationPeriod
- Complex Type: Variance
- Model Group: CurrencyAndDeterminationMethod.model
- Model Group: DeclaredCashAndCashEquivalentDividendPercentage.model
- Model Group: EquityUnderlyerProvisions.model
- Model Group: Feature.model
- Model Group: IndexAnnexFallback.model
- Model Group: MutualOrOptionalEarlyTermination.model

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 7132 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-option-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 7132 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-option-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
```

```
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the

substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

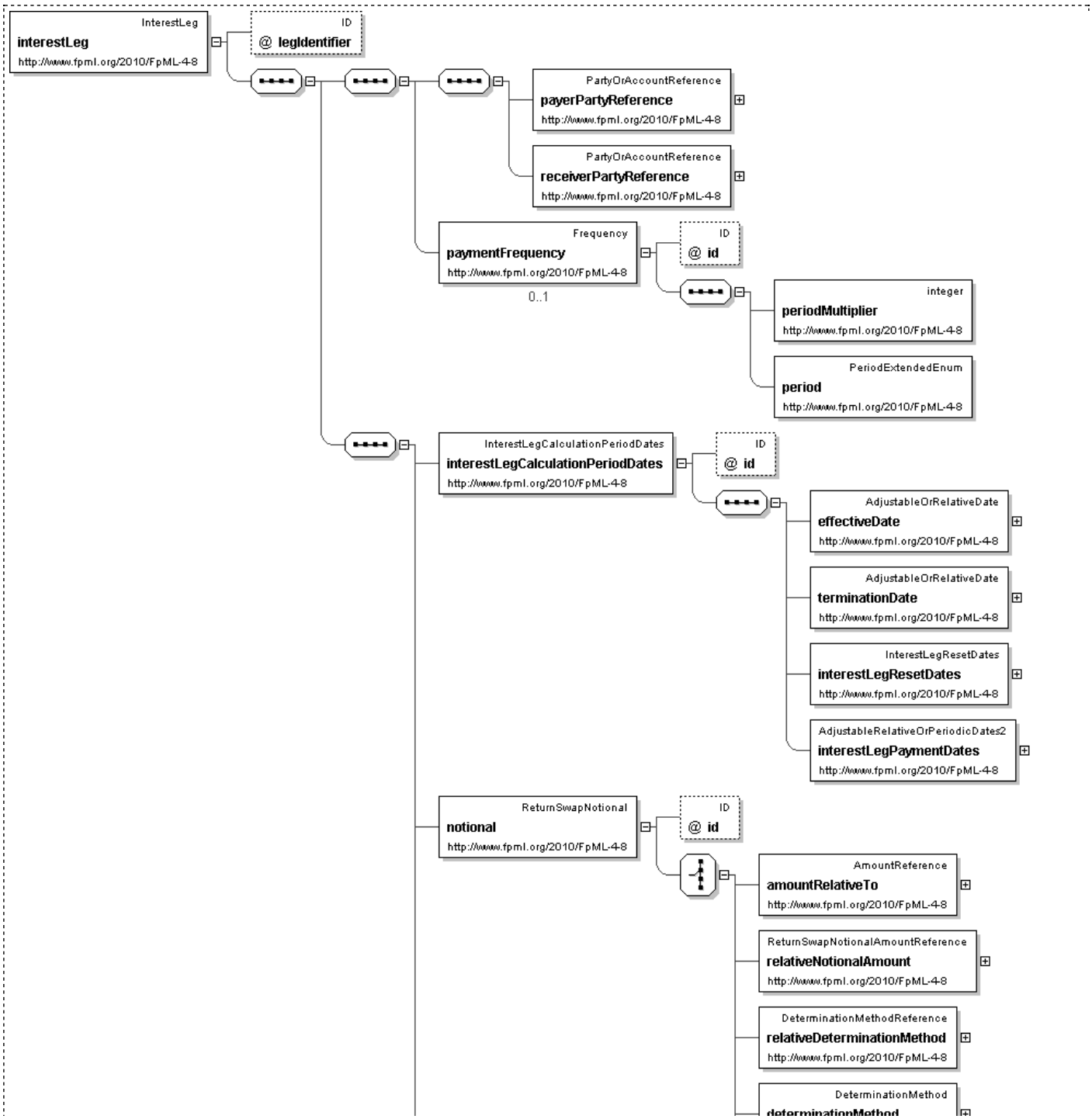
Element: interestLeg

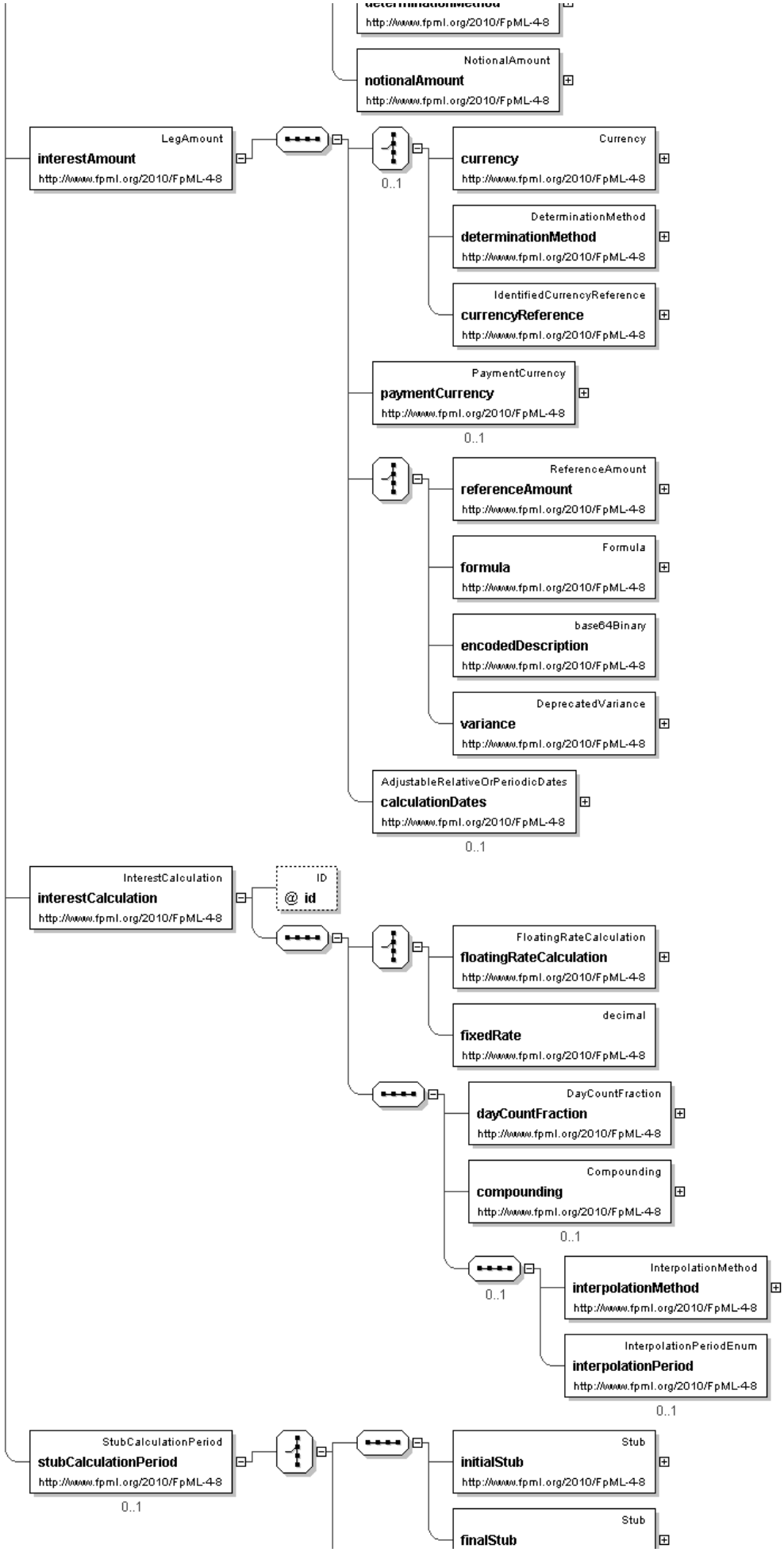
[Table of contents]

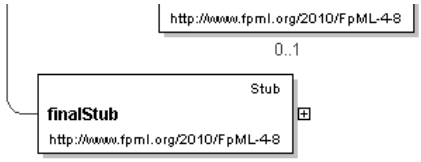
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	interestLeg
Type	InterestLeg
Nillable	no
Abstract	no
Documentation	The fixed income amounts of the return type swap.

Logical Diagram



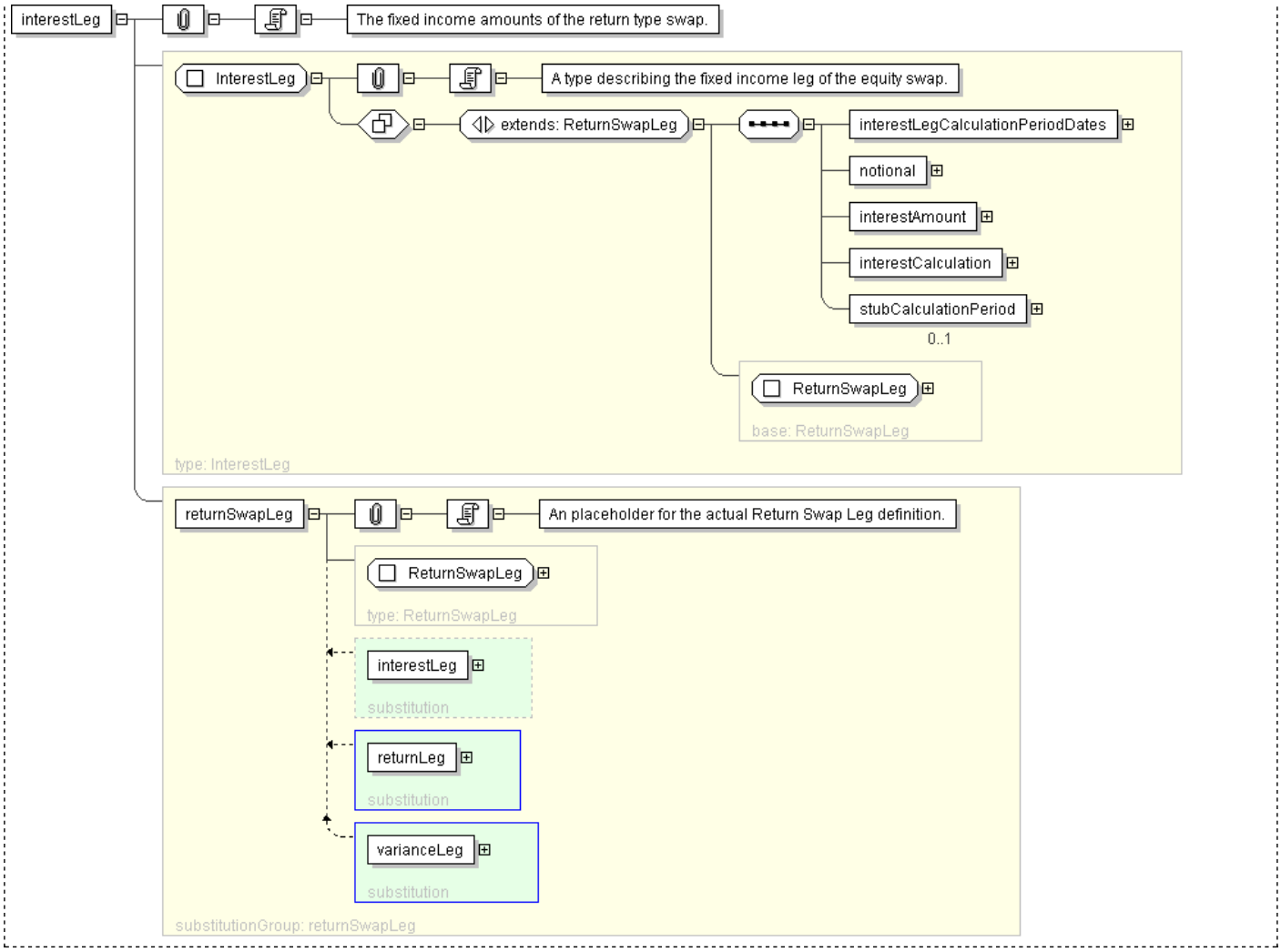

http://www.fpml.org/2010/FpML-4-8
NotionalAmount
notionalAmount
http://www.fpml.org/2010/FpML-4-8
LegAmount
interestAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2010/FpML-4-8
PaymentCurrency
paymentCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2010/FpML-4-8
Formula
formula
http://www.fpml.org/2010/FpML-4-8
base64Binary
encodedDescription
http://www.fpml.org/2010/FpML-4-8
DeprecatedVariance
variance
http://www.fpml.org/2010/FpML-4-8
AdjustableRelativeOrPeriodicDates
calculationDates
http://www.fpml.org/2010/FpML-4-8
0..1
InterestCalculation
interestCalculation
http://www.fpml.org/2010/FpML-4-8
ID
@ id
FloatingRateCalculation
floatingRateCalculation
http://www.fpml.org/2010/FpML-4-8
decimal
fixedRate
http://www.fpml.org/2010/FpML-4-8
DayCountFraction
dayCountFraction
http://www.fpml.org/2010/FpML-4-8
Compounding
compounding
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
InterpolationMethod
interpolationMethod
http://www.fpml.org/2010/FpML-4-8
InterpolationPeriodEnum
interpolationPeriod
http://www.fpml.org/2010/FpML-4-8
0..1
StubCalculationPeriod
stubCalculationPeriod
http://www.fpml.org/2010/FpML-4-8
0..1
Stub
initialStub
http://www.fpml.org/2010/FpML-4-8
Stub
finalStub



XML Instance Representation

```
<interestLeg
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <interestLegCalculationPeriodDates> InterestLegCalculationPeriodDates </interestLegCalculationPeriodDates> [1]
   'Component that holds the various dates used to specify the interest leg of the equity swap. It is used to define the
   InterestPeriodDates identifyer.'

   <notional> ReturnSwapNotional </notional> [1]
   'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine the
   actual amount (using the notional component defined by the FpML industry group) and the determination method. When
   used in the interest leg, the definition will typically point to the definition of the equity leg.'

   <interestAmount> LegAmount </interestAmount> [1]
   'Specifies, in relation to each Interest Payment Date, the amount to which the Interest Payment Date relates. Unless
   otherwise specified, this term has the meaning defined in the ISDA 2000 ISDA Definitions.'

   <interestCalculation> InterestCalculation </interestCalculation> [1]
   'Specifies the calculation method of the interest rate leg of the equity swap. Includes the floating or fixed rate
   calculation definitions, along with the determination of the day count fraction.'

   <stubCalculationPeriod> StubCalculationPeriod </stubCalculationPeriod> [0..1]
   'Specifies the stub calculation period.'

</interestLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="interestLeg" type=" InterestLeg " substitutionGroup="returnSwapLeg"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

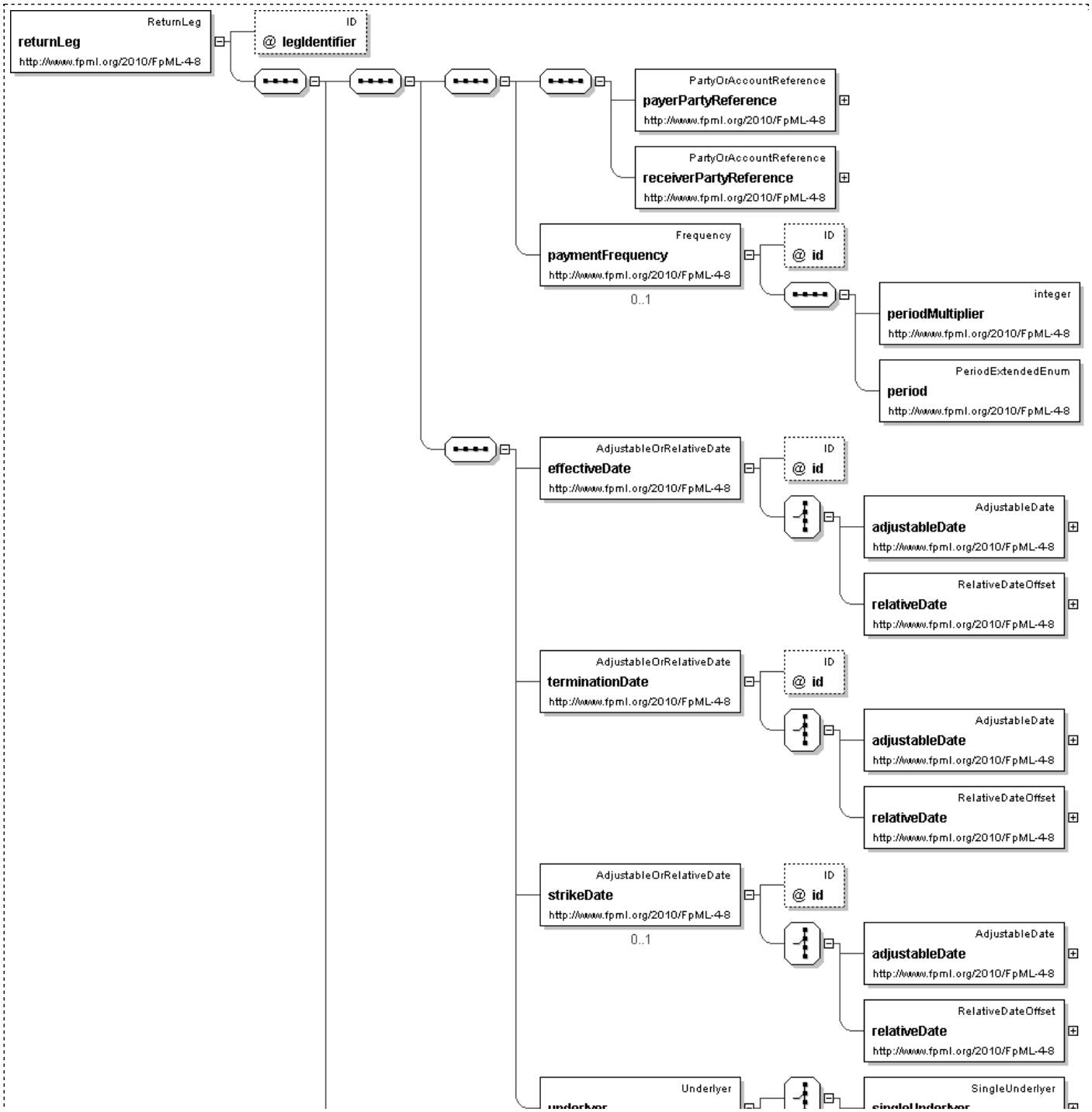
Element: returnLeg

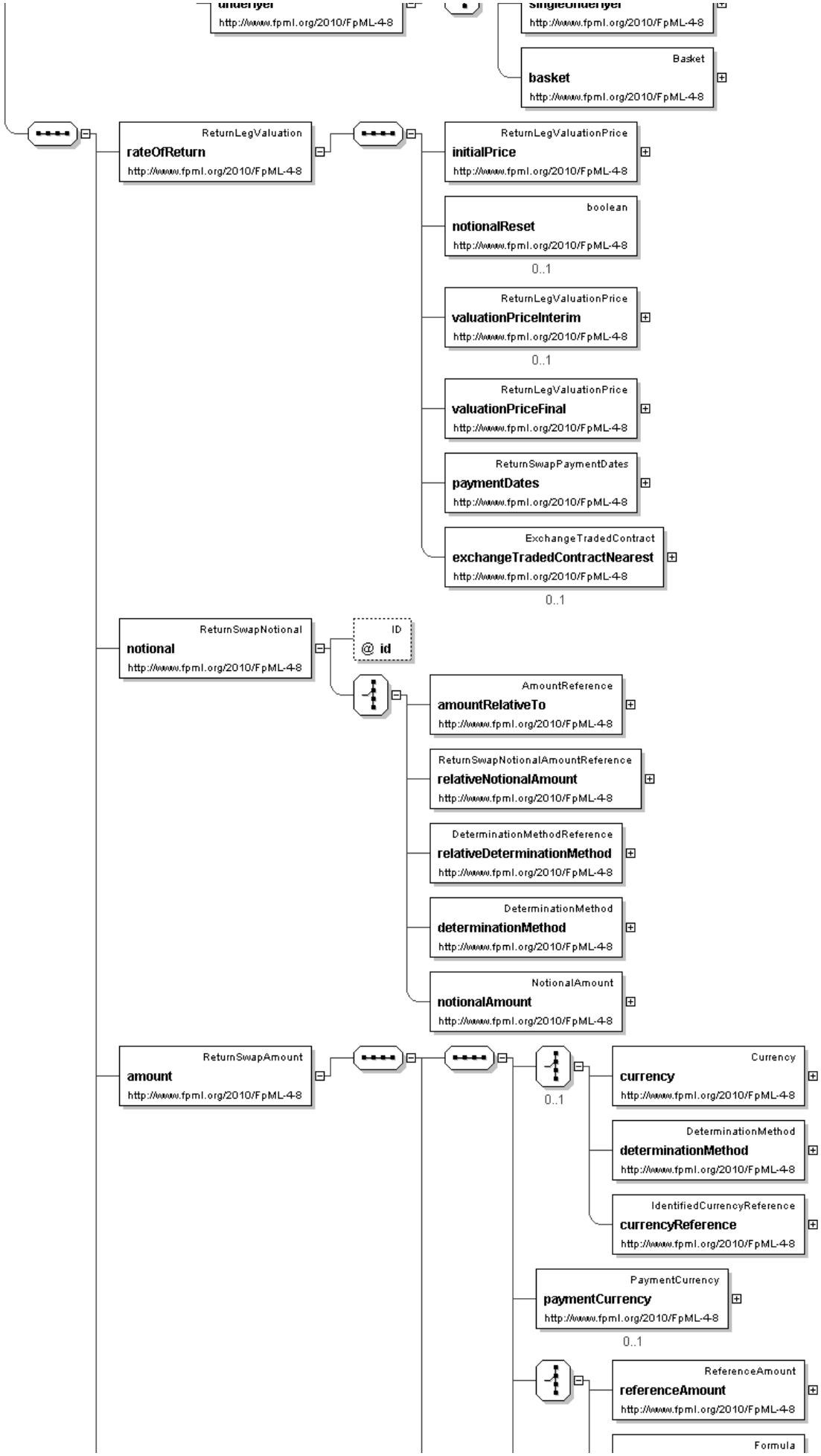
[Table of contents]

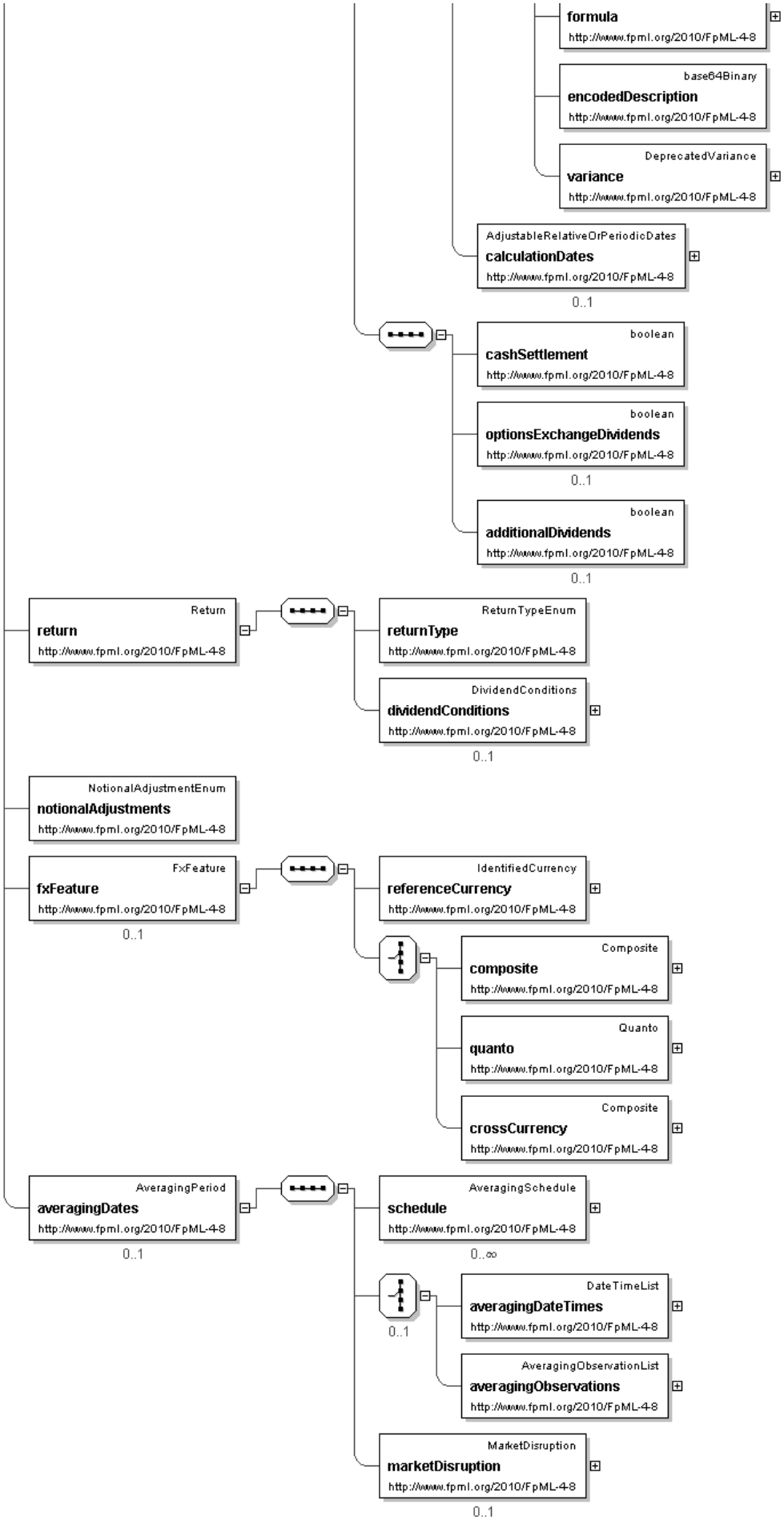
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	returnLeg
Type	ReturnLeg
Nillable	no
Abstract	no
Documentation	Return amounts of the return type swap.

Logical Diagram



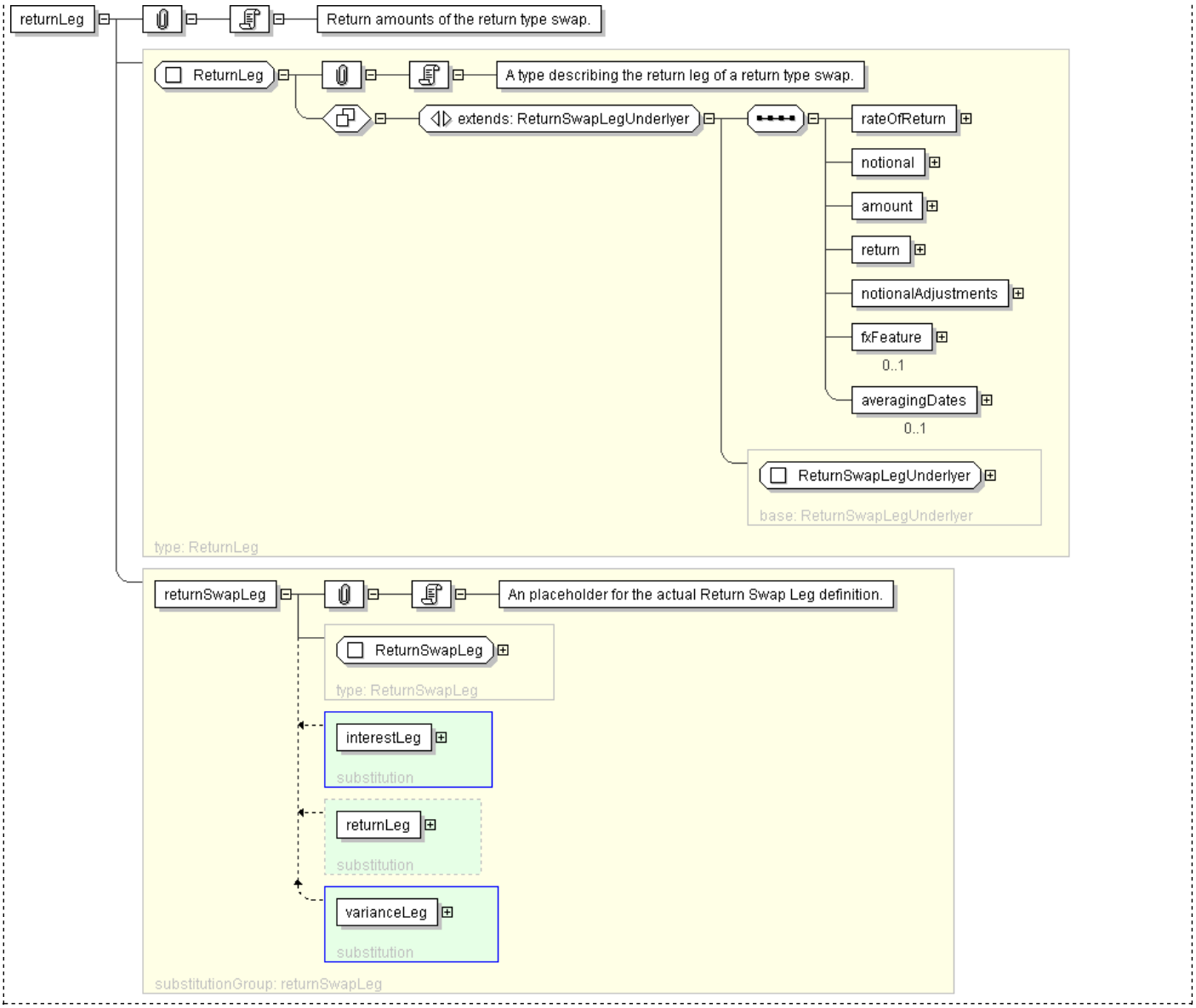

underlyer
http://www.fpml.org/2010/FpML-4-8
singleUnderlyer
http://www.fpml.org/2010/FpML-4-8
Basket
basket
http://www.fpml.org/2010/FpML-4-8
ReturnLegValuation
rateOfReturn
http://www.fpml.org/2010/FpML-4-8
ReturnLegValuationPrice
initialPrice
http://www.fpml.org/2010/FpML-4-8
boolean
notionalReset
http://www.fpml.org/2010/FpML-4-8
0..1
ReturnLegValuationPrice
valuationPriceInterim
http://www.fpml.org/2010/FpML-4-8
0..1
ReturnLegValuationPrice
valuationPriceFinal
http://www.fpml.org/2010/FpML-4-8
ReturnSwapPaymentDates
paymentDates
http://www.fpml.org/2010/FpML-4-8
ExchangeTradedContract
exchangeTradedContractNearest
http://www.fpml.org/2010/FpML-4-8
0..1
ReturnSwapNotional
notional
http://www.fpml.org/2010/FpML-4-8
ID
@ id
AmountReference
amountRelativeTo
http://www.fpml.org/2010/FpML-4-8
ReturnSwapNotionalAmountReference
relativeNotionalAmount
http://www.fpml.org/2010/FpML-4-8
DeterminationMethodReference
relativeDeterminationMethod
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
NotionalAmount
notionalAmount
http://www.fpml.org/2010/FpML-4-8
ReturnSwapAmount
amount
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2010/FpML-4-8
PaymentCurrency
paymentCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2010/FpML-4-8
Formula



XML Instance Representation

```
<returnLeg
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere else
   in the document (through the relativeDate component), this element will typically point to the effective date of
   the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere
   else in the document (through the relativeDate component), this element will typically point to the termination date
   of the other leg of the swap.'

   <strikeDate> AdjustableOrRelativeDate </strikeDate> [0..1]
   'Specifies the strike date of this leg of the swap, used for forward starting swaps. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component), this element will typically by
   relative to the trade date of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component of the leg, which can be either one or many and consists in either equity,
   index or convertible bond component, or a combination of these.'

   <rateOfReturn> ReturnLegValuation </rateOfReturn> [1]
   'Element named \"valuation\" in versions prior to FpML 4.2 Second Working Draft. Specifies the terms of the initial
   price of the return type swap and of the subsequent valuations of the underlyer.'

   <notional> ReturnSwapNotional </notional> [1]
   'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine
   the actual amount (using the notional component defined by the FpML industry group) and the determination method.
   When used in the interest leg, the definition will typically point to the definition of the equity leg.'

   <amount> ReturnSwapAmount </amount> [1]
   'Element named \"equityAmount\" in versions prior to FpML 4.2 Second Working Draft. Specifies, in relation to each
   Payment Date, the amount to which the Payment Date relates. For equity swaps this element is equivalent to the
   Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.'

   <return> Return </return> [1]
   'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the
   amounts.'

   <notionalAdjustments> NotionalAdjustmentEnum </notionalAdjustments> [1]
   'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

   <fxFeature> FxFeature </fxFeature> [0..1]
   'A quanto or composite FX feature.'

   <averagingDates> AveragingPeriod </averagingDates> [0..1]
   'Averaging Dates used in the swap.'

</returnLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="returnLeg" type=" ReturnLeg " substitutionGroup="returnSwapLeg"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

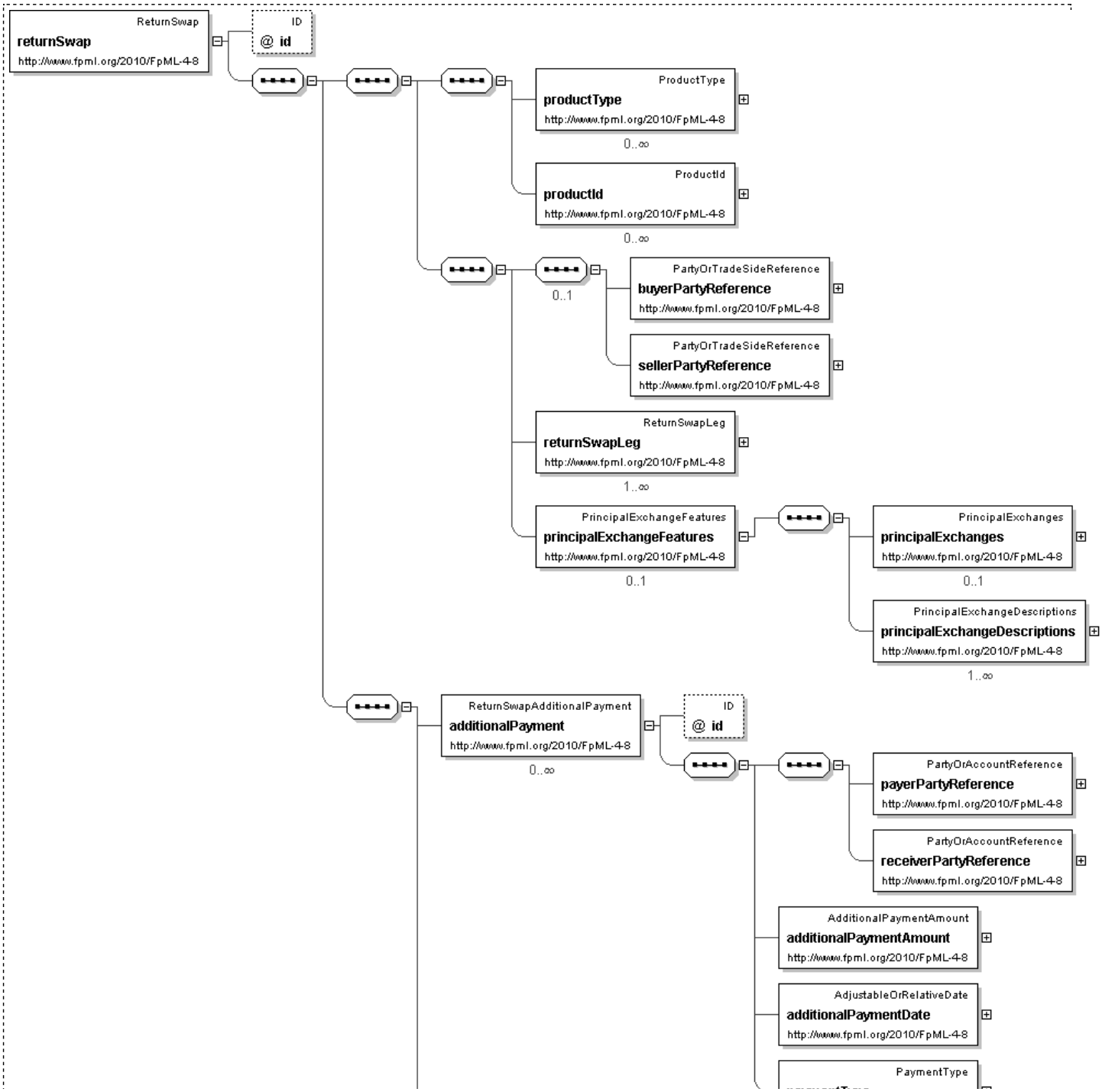
Element: returnSwap

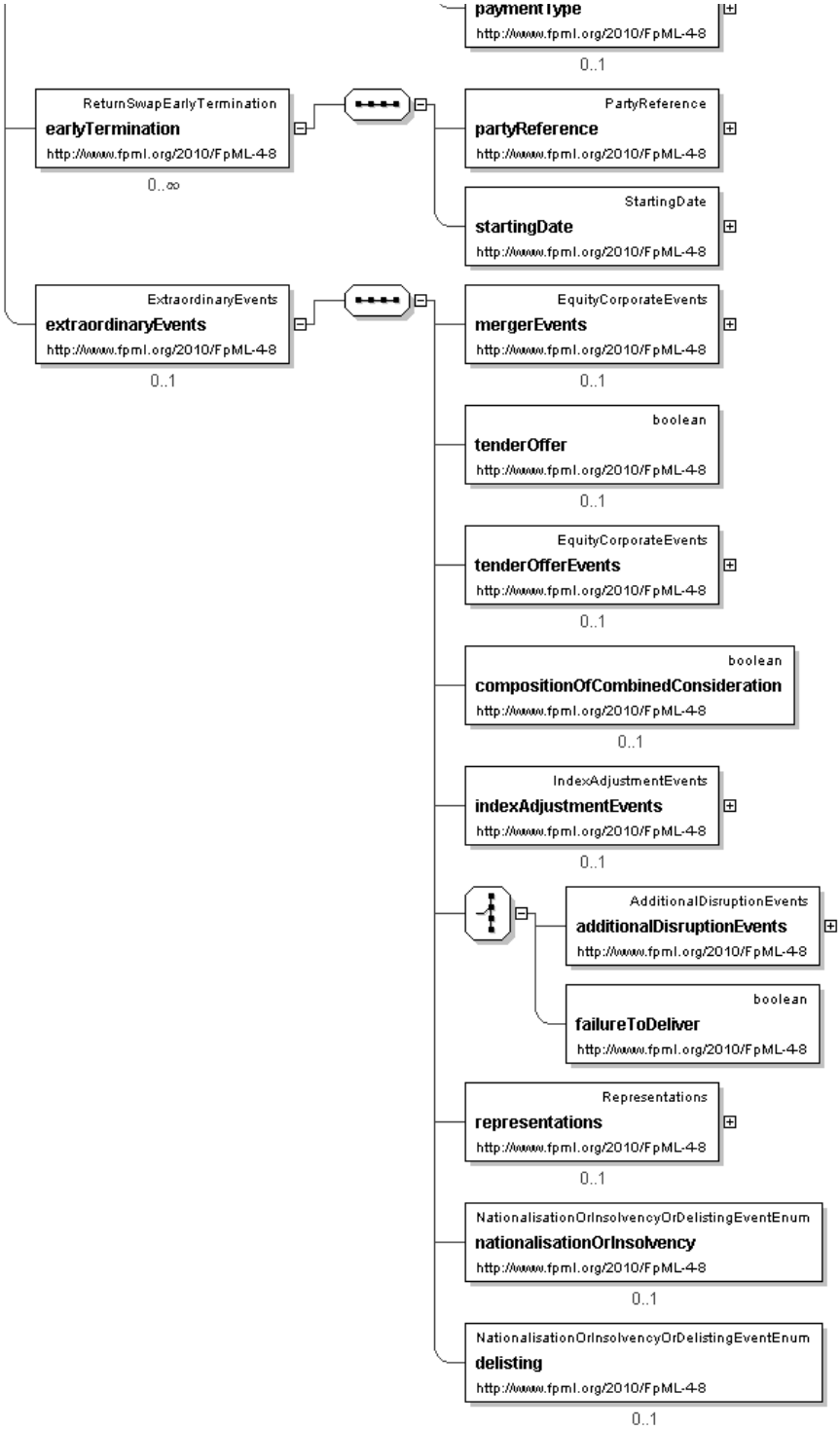
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	returnSwap
Type	ReturnSwap
Nillable	no
Abstract	no
Documentation	Specifies the structure of a return type swap. It can represent equity swaps, total return swaps, variance swaps.

Logical Diagram



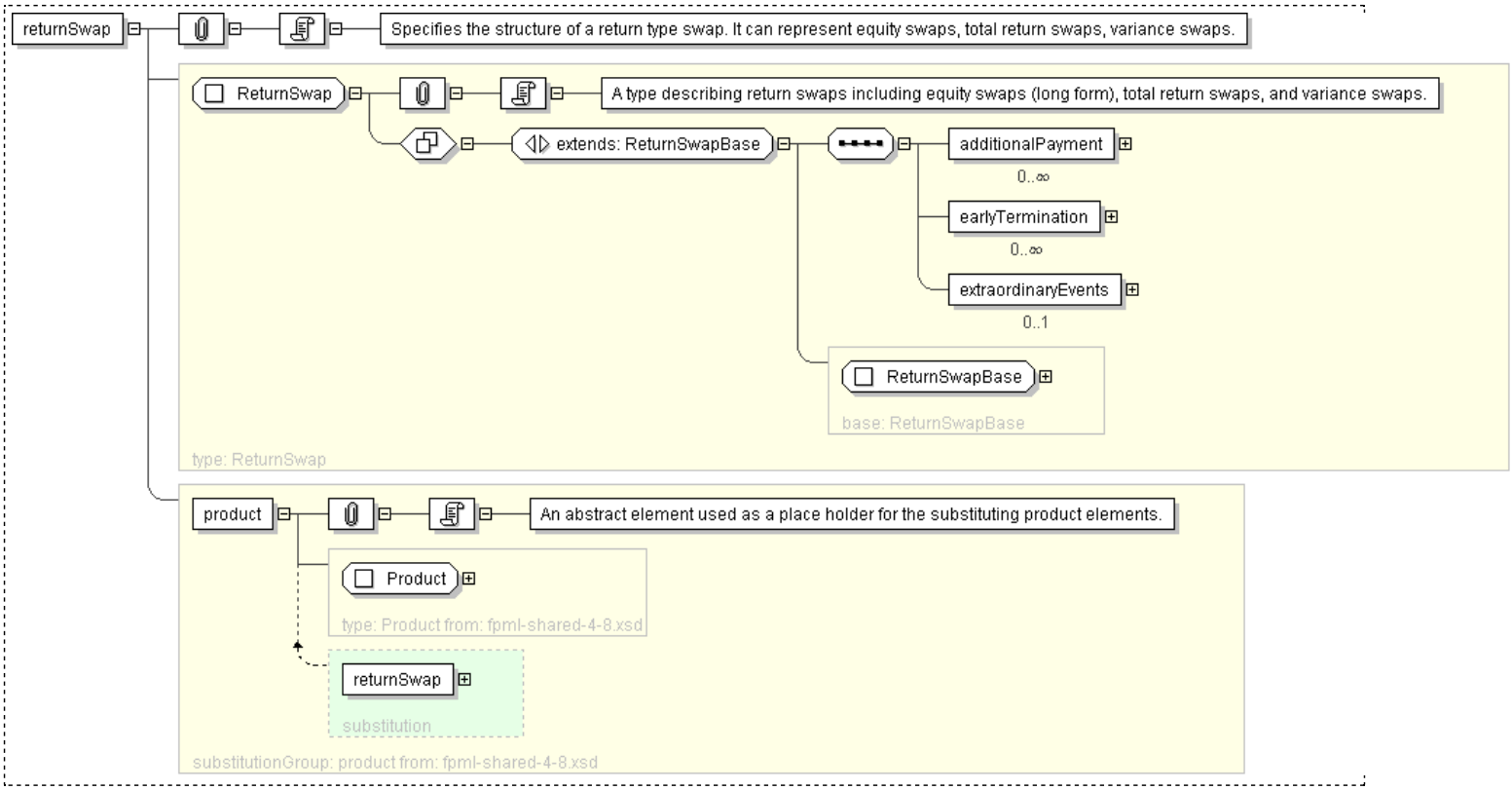


XML Instance Representation

```
<returnSwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this
   element. Note that the domain values for this element are not strictly an enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where an
implementor wishes to indicate who has manufactured the Swap through representing them as the Seller. It may be
removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
      defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this
      instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of
```

```
    FRAs this is the floating rate payer.'

  End Group: BuyerSeller.model
  <returnSwapLeg> ... </returnSwapLeg> [1..*]
  <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
  'This is used to document a Fully Funded Return Swap.'

  <additionalPayment> ReturnSwapAdditionalPayment </additionalPayment> [0..*]
  'Specifies additional payment(s) between the principal parties to the trade. This component extends some of the
  features of the additionalPayment component developed by the FpML industry group. Appropriate discussions will
  determine whether it would be appropriate to extend the shared component in order to meet the further requirements
  of equity swaps.'

  <earlyTermination> ReturnSwapEarlyTermination </earlyTermination> [0..*]
  'Specifies, for one or for both the parties to the trade, the date from which it can early terminate it.'

  <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
  'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms
  of the transaction to be adjusted.'

</returnSwap>
```

Diagram



Schema Component Representation

```
<xsd:element name="returnSwap" type=" ReturnSwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

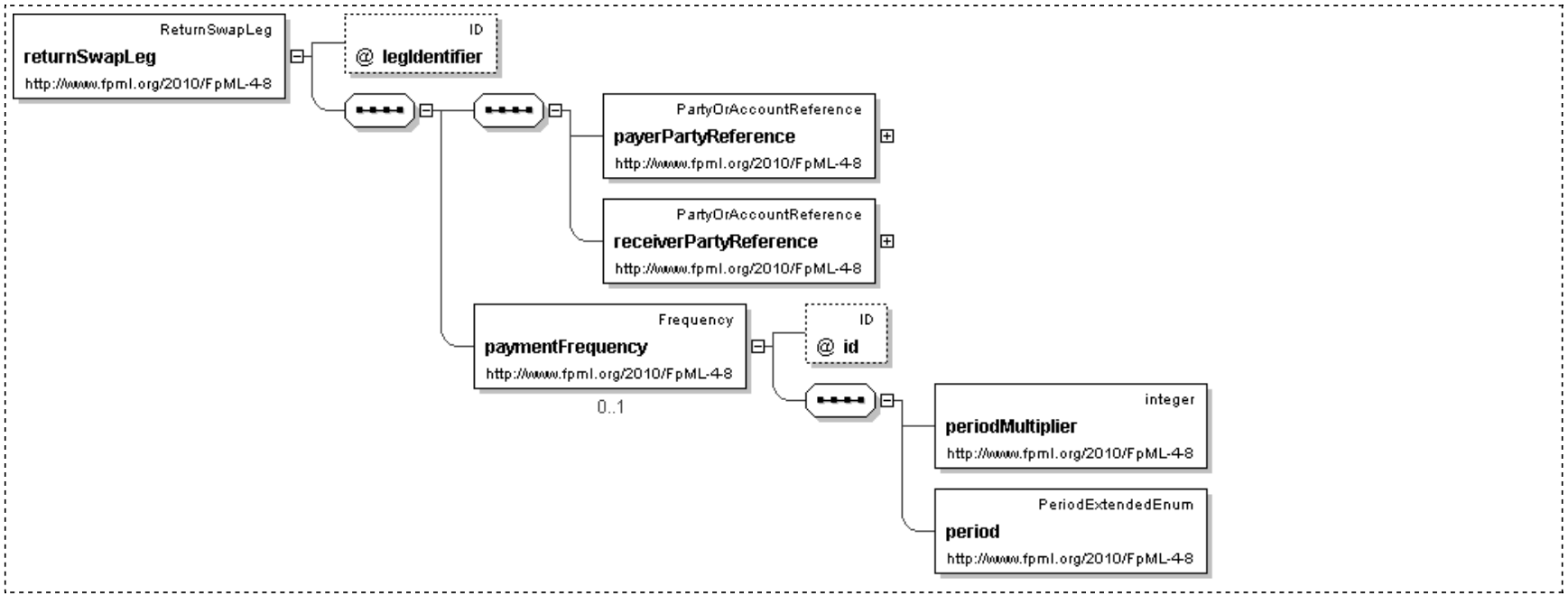
Element: returnSwapLeg

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - interestLeg
  - returnLeg
  - varianceLeg

Name	returnSwapLeg
Used by (from the same schema document)	Complex Type ReturnSwapBase
Type	ReturnSwapLeg
Nillable	no
Abstract	yes
Documentation	An placeholder for the actual Return Swap Leg definition.

Logical Diagram

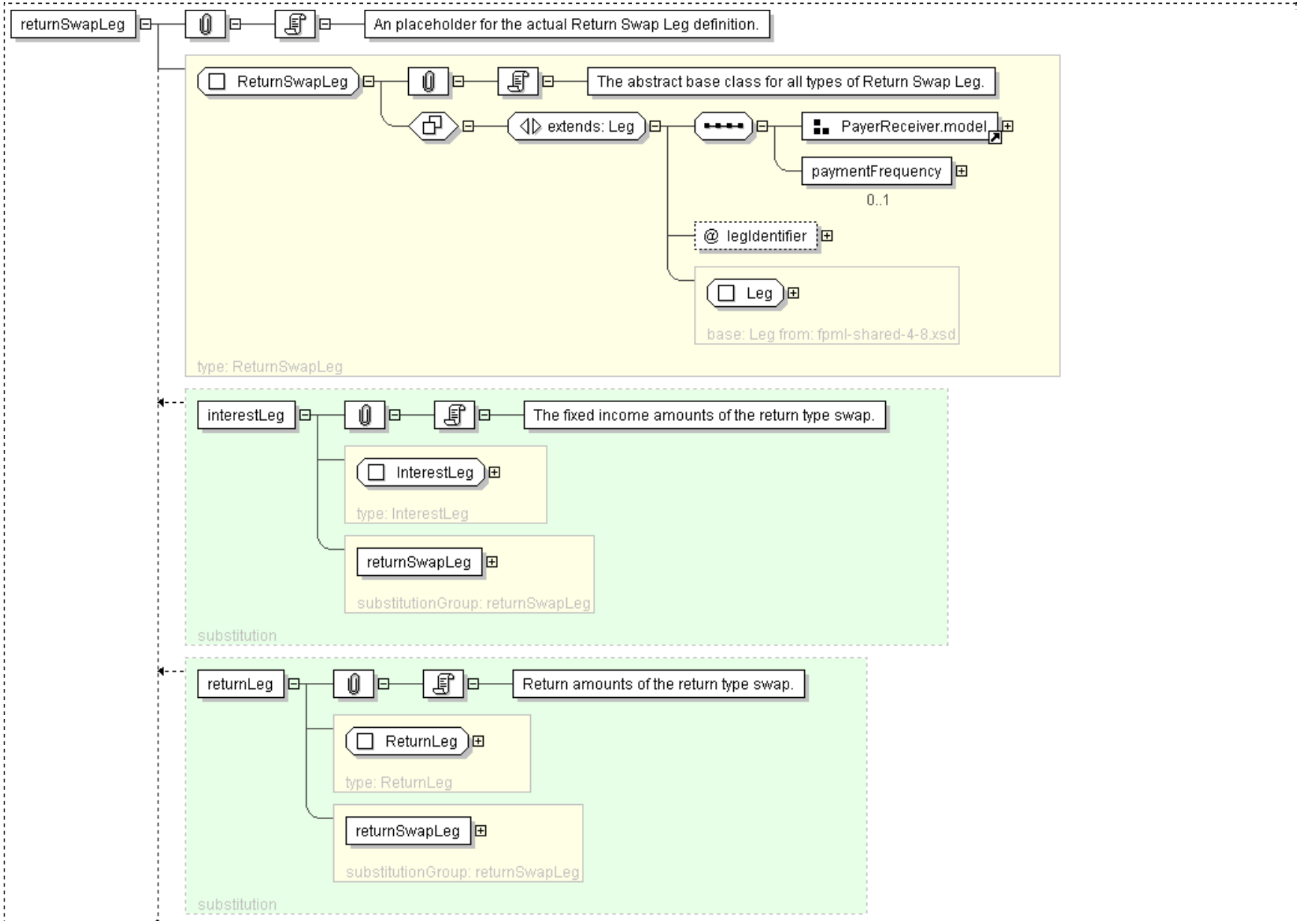


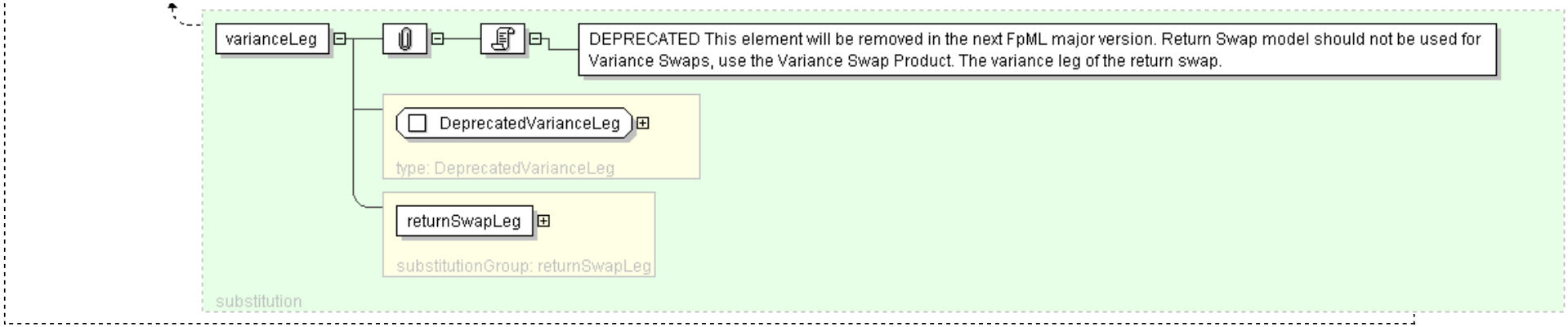
XML Instance Representation

```
<returnSwapLeg
 legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'
```

```
">
  <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
  'A reference to the party responsible for making the payments defined by this structure.'

  <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
  'A reference to the party that receives the payments corresponding to this structure.'

  <paymentFrequency> Frequency </paymentFrequency> [0..1]
  'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

</returnSwapLeg>
```

Diagram





Schema Component Representation

```
<xsd:element name="returnSwapLeg" type=" ReturnSwapLeg " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

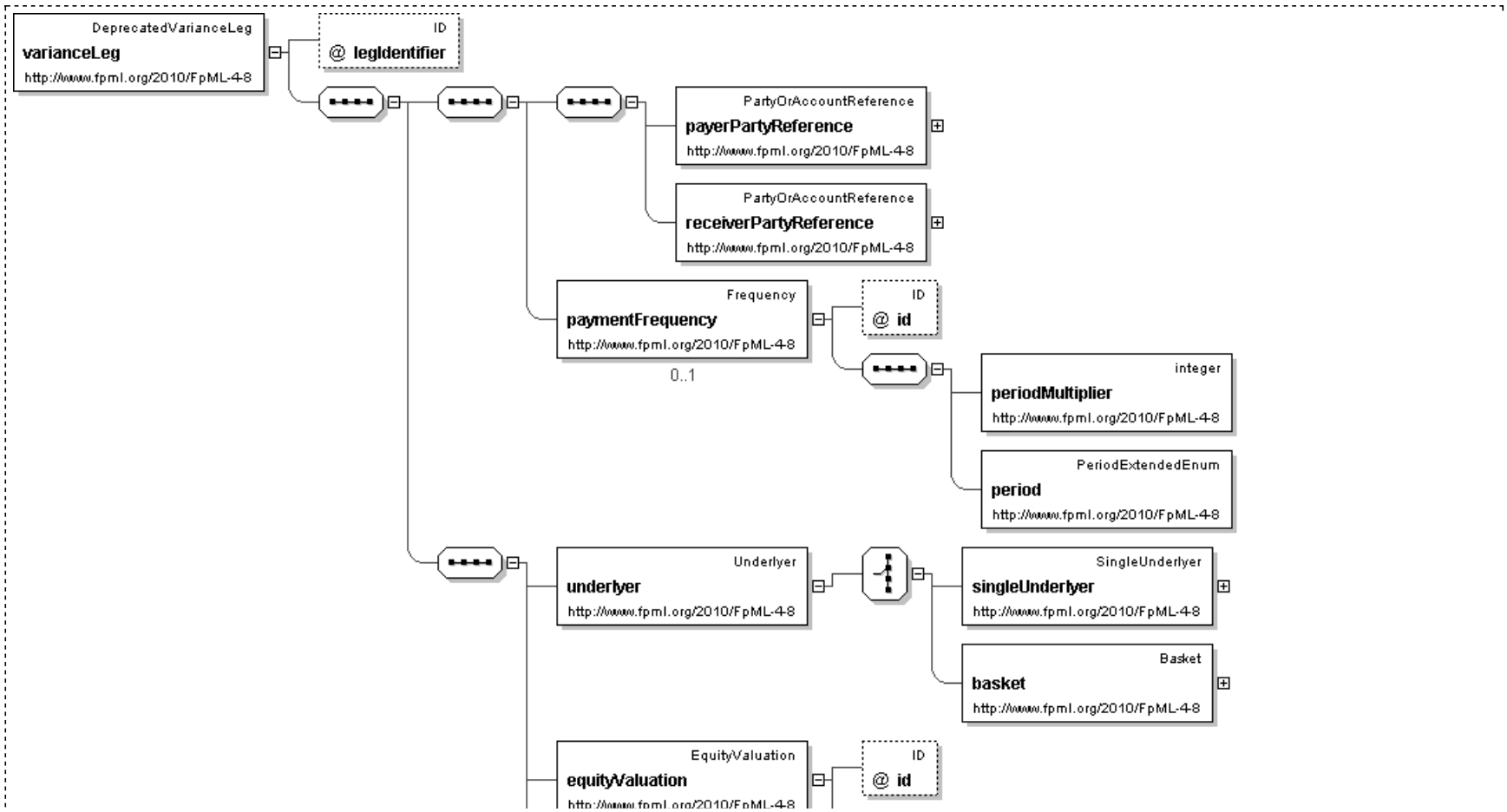
Element: varianceLeg

[Table of contents]

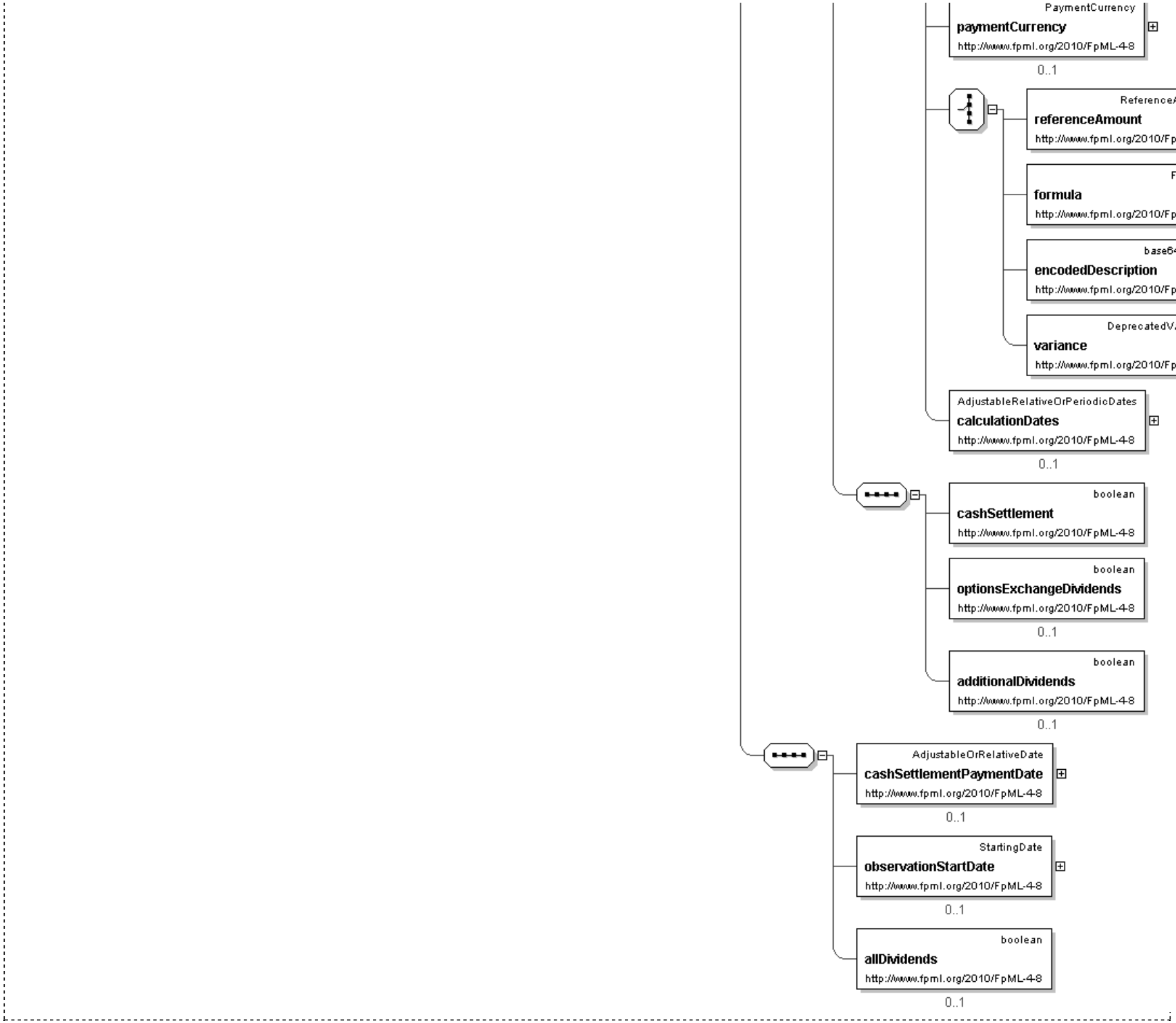
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	varianceLeg
Type	DeprecatedVarianceLeg
Nillable	no
Abstract	no
Documentation	DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. The variance leg of the return swap.

Logical Diagram



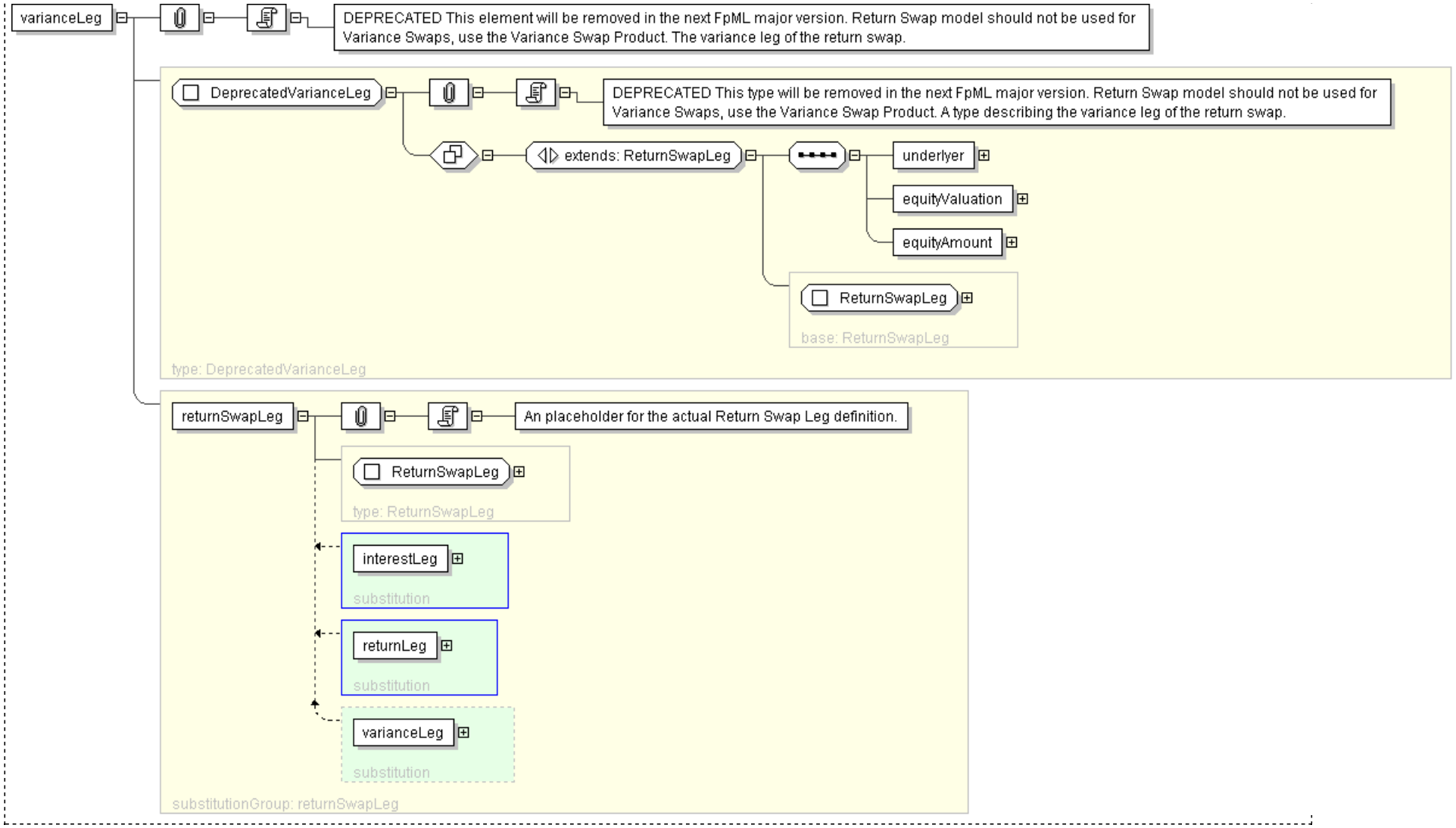
0..1
AdjustableDateOrRelativeDateSequence
valuationDate
http://www.fpml.org/2010/FpML-4-8
AdjustableRelativeOrPeriodicDates
valuationDates
http://www.fpml.org/2010/FpML-4-8
TimeTypeEnum
valuationTimeType
http://www.fpml.org/2010/FpML-4-8
0..1
BusinessCenterTime
valuationTime
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
futuresPriceValuation
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
optionsPriceValuation
http://www.fpml.org/2010/FpML-4-8
0..1
nonNegativeInteger
numberOfValuationDates
http://www.fpml.org/2010/FpML-4-8
0..1
AdjustableRelativeOrPeriodicDates
dividendValuationDates
http://www.fpml.org/2010/FpML-4-8
0..1
FPVFinalPriceElectionFallbackEnum
fPVFinalPriceElectionFallback
http://www.fpml.org/2010/FpML-4-8
0..1
DeprecatedVarianceAmount
equityAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2010/FpML-4-8


PaymentCurrency
paymentCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2010/FpML-4-8
Formula
formula
http://www.fpml.org/2010/FpML-4-8
base64Binary
encodedDescription
http://www.fpml.org/2010/FpML-4-8
DeprecatedVariance
variance
http://www.fpml.org/2010/FpML-4-8
AdjustableRelativeOrPeriodicDates
calculationDates
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
cashSettlement
http://www.fpml.org/2010/FpML-4-8
boolean
optionsExchangeDividends
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
additionalDividends
http://www.fpml.org/2010/FpML-4-8
0..1
AdjustableOrRelativeDate
cashSettlementPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
StartingDate
observationStartDate
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
allDividends
http://www.fpml.org/2010/FpML-4-8
0..1

XML Instance Representation

```
<varianceLeg
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <equityValuation> EquityValuation </equityValuation> [1]
   'Valuation of the underlyer.'

   <equityAmount> DeprecatedVarianceAmount </equityAmount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates. Unless otherwise
   specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.'

</varianceLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="varianceLeg" type=" DeprecatedVarianceLeg " substitutionGroup="returnSwapLeg" deprecated="true"
deprecatedReason="Return Swap model should not be used for Variance Swaps, use the Variance Swap Product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CurrencyAndDeterminationMethod.model

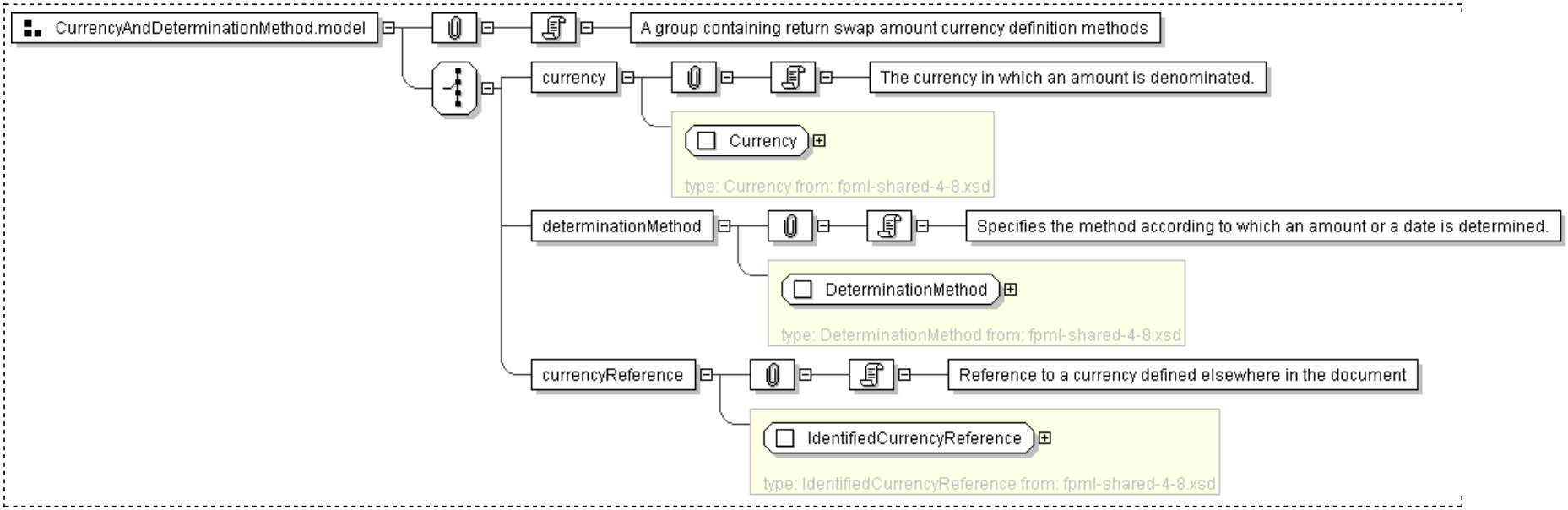
[Table of contents]

Name	CurrencyAndDeterminationMethod.model
Used by (from the same schema document)	Complex Type DividendConditions , Complex Type LegAmount
Documentation	A group containing return swap amount currency definition methods

XML Instance Representation

```
Start Choice [1]
   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

   <determinationMethod> DeterminationMethod </determinationMethod> [1]
   'Specifies the method according to which an amount or a date is determined.'

   <currencyReference> IdentifiedCurrencyReference </currencyReference> [1]
   'Reference to a currency defined elsewhere in the document'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="CurrencyAndDeterminationMethod.model">
   <xsd:choice>
      <xsd:element name="currency" type=" Currency "/>
      <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
      <xsd:element name="currencyReference" type=" IdentifiedCurrencyReference "/>
   </xsd:choice>
```

```
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: DeclaredCashAndCashEquivalentDividendPercentage.model

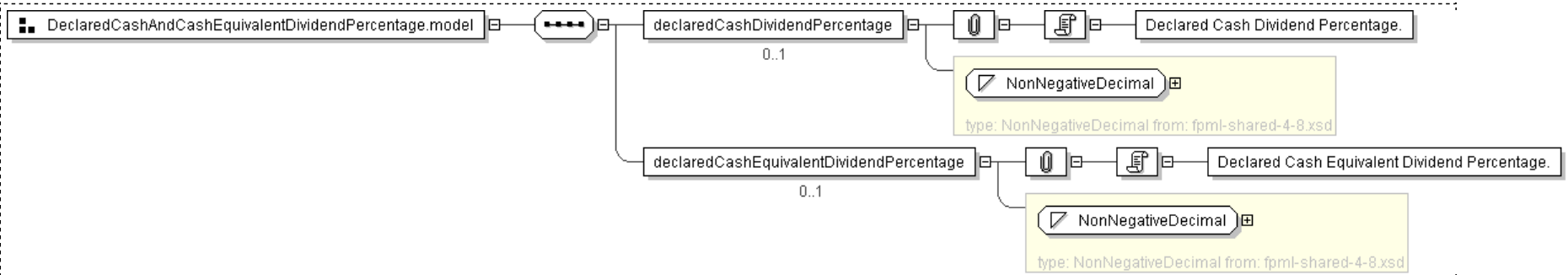
[Table of contents]

Name	DeclaredCashAndCashEquivalentDividendPercentage.model
Used by (from the same schema document)	Complex Type DividendConditions

XML Instance Representation

```
<declaredCashDividendPercentage> NonNegativeDecimal </declaredCashDividendPercentage> [0..1]
'Declared Cash Dividend Percentage.'

<declaredCashEquivalentDividendPercentage> NonNegativeDecimal </declaredCashEquivalentDividendPercentage> [0..1]
'Declared Cash Equivalent Dividend Percentage.'
```

Diagram



Schema Component Representation

```
<xsd:group name="DeclaredCashAndCashEquivalentDividendPercentage.model">
  <xsd:sequence>
    <xsd:element name="declaredCashDividendPercentage" type=" NonNegativeDecimal " minOccurs="0"/>
    <xsd:element name="declaredCashEquivalentDividendPercentage" type=" NonNegativeDecimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: EquityUnderlyerProvisions.model

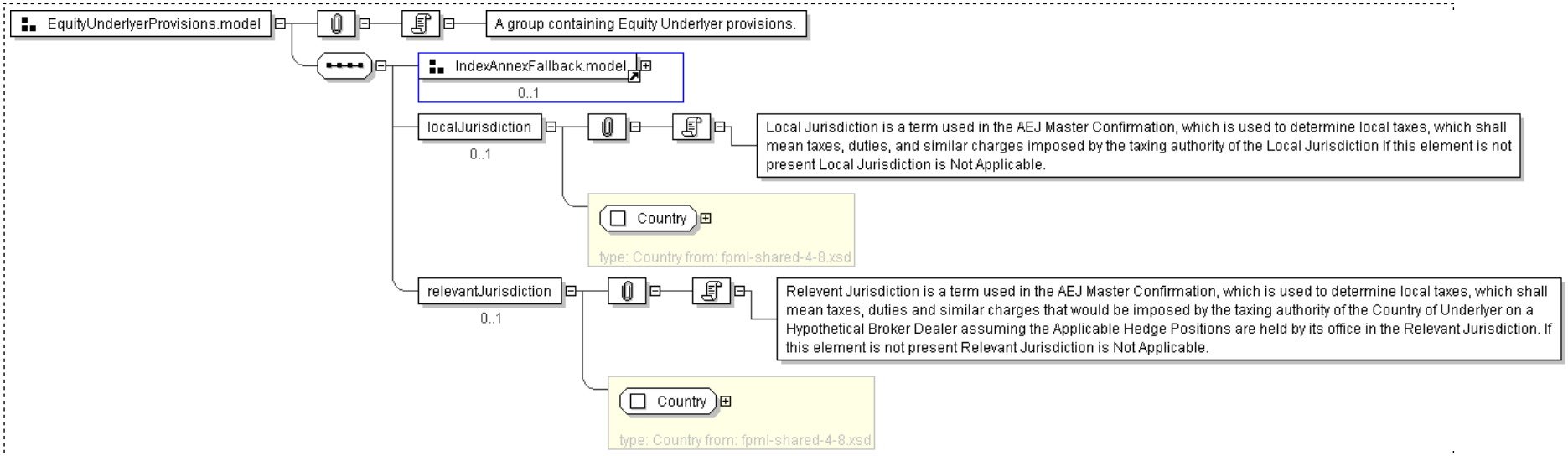
[Table of contents]

Name	EquityUnderlyerProvisions.model
Documentation	A group containing Equity Underlyer provisions.

XML Instance Representation

```
Start Group: IndexAnnexFallback.model [0..1]
  Start Choice [1]
    <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
    'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the transaction. This
    annex defines additional provisions which are applicable where an index is comprised of component securities that are traded on multiple
    exchanges.'

    <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
    'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the transaction.'

  End Choice
End Group: IndexAnnexFallback.model
<localJurisdiction> Country </localJurisdiction> [0..1]
'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes, duties, and similar
charges imposed by the taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction is Not Applicable.'

<relevantJurisdiction> Country </relevantJurisdiction> [0..1]
'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes, duties and
similar charges that would be imposed by the taxing authority of the Country of Underlyer on a Hypothetical Broker Dealer assuming the Applicable
Hedge Positions are held by its office in the Relevant Jurisdiction. If this element is not present Relevant Jurisdiction is Not Applicable.'
```

Diagram



Schema Component Representation

```
<xsd:group name="EquityUnderlyerProvisions.model">
  <xsd:sequence>
    <xsd:group ref=" IndexAnnexFallback.model " minOccurs="0"/>
    <xsd:element name="localJurisdiction" type=" Country " minOccurs="0"/>
    <xsd:element name="relevantJurisdiction" type=" Country " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Feature.model

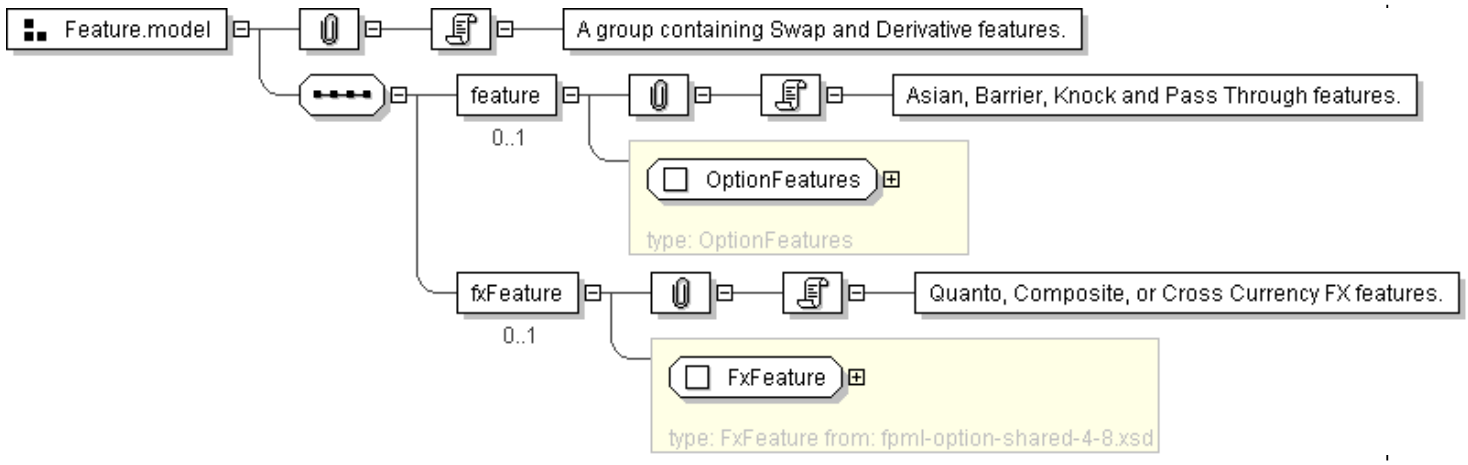
[Table of contents]

Name	Feature.model
Documentation	A group containing Swap and Derivative features.

XML Instance Representation

```
<feature> OptionFeatures </feature> [0..1]
'Asian, Barrier, Knock and Pass Through features.'

<fxFeature> FxFeature </fxFeature> [0..1]
'Quanto, Composite, or Cross Currency FX features.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Feature.model">
   <xsd:sequence>
      <xsd:element name="feature" type=" OptionFeatures " minOccurs="0"/>
      <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: IndexAnnexFallback.model

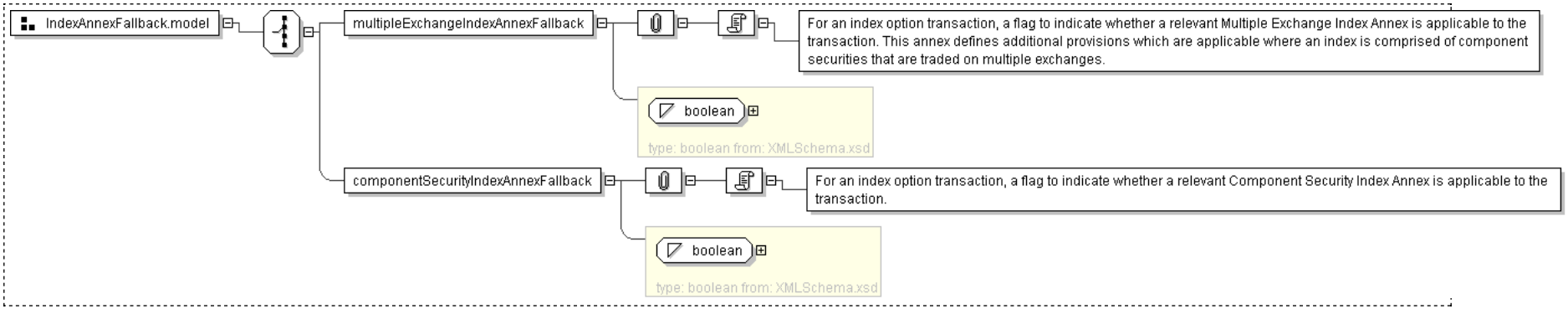
[Table of contents]

Name	IndexAnnexFallback.model
Used by (from the same schema document)	Model Group EquityUnderlyerProvisions.model

XML Instance Representation

```
Start Choice [1]
   <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
   'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the transaction. This annex defines
   additional provisions which are applicable where an index is comprised of component securities that are traded on multiple exchanges.'

   <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
   'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the transaction.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="IndexAnnexFallback.model">
   <xsd:choice>
      <xsd:element name="multipleExchangeIndexAnnexFallback" type=" xsd:boolean "/>
      <xsd:element name="componentSecurityIndexAnnexFallback" type=" xsd:boolean "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: MutualOrOptionalEarlyTermination.model

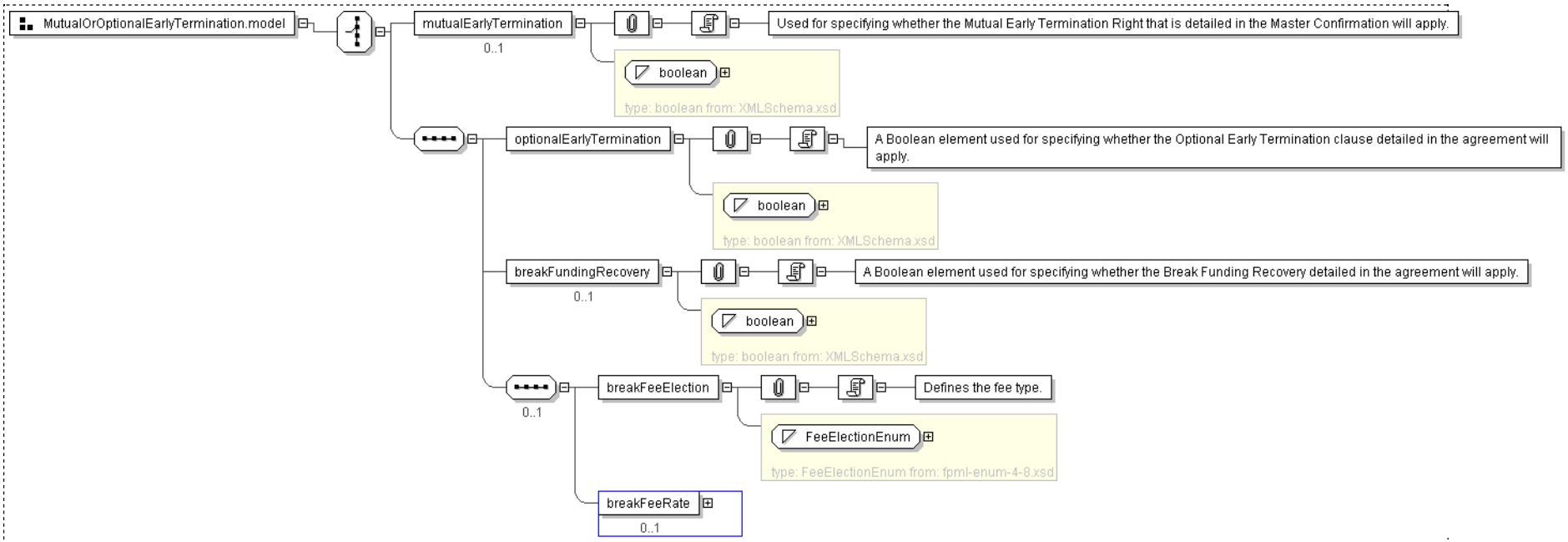
[Table of contents]

Name	MutualOrOptionalEarlyTermination.model

XML Instance Representation

```
Start Choice [1]
   <mutualEarlyTermination> xsd:boolean </mutualEarlyTermination> [0..1]
   'Used for specifying whether the Mutual Early Termination Right that is detailed in the Master Confirmation will apply.'

   <optionalEarlyTermination> xsd:boolean </optionalEarlyTermination> [1]
   'A Boolean element used for specifying whether the Optional Early Termination clause detailed in the agreement will apply.'

   <breakFundingRecovery> xsd:boolean </breakFundingRecovery> [0..1]
   'A Boolean element used for specifying whether the Break Funding Recovery detailed in the agreement will apply.'

   Start Sequence [0..1]
      <breakFeeElection> FeeElectionEnum </breakFeeElection> [1]
      'Defines the fee type.'

      <breakFeeRate> NonNegativeDecimal </breakFeeRate> [0..1]
   End Sequence
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="MutualOrOptionalEarlyTermination.model">
   <xsd:choice>
      <xsd:element name="mutualEarlyTermination" type=" xsd:boolean " minOccurs="0"/>
      <xsd:sequence>
         <xsd:element name="optionalEarlyTermination" type=" xsd:boolean "/>
         <xsd:element name="breakFundingRecovery" type=" xsd:boolean " minOccurs="0"/>
         <xsd:sequence minOccurs="0">
            <xsd:element name="breakFeeElection" type=" FeeElectionEnum "/>
            <xsd:element name="breakFeeRate" type=" NonNegativeDecimal " minOccurs="0"/>
         </xsd:sequence>
      </xsd:sequence>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdditionalDisruptionEvents

[Table of contents]

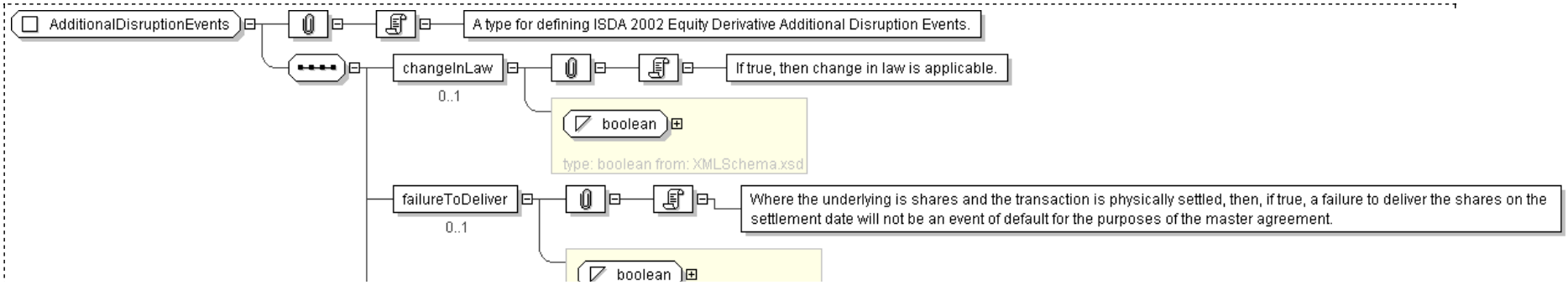
Super-types:	None
Sub-types:	None

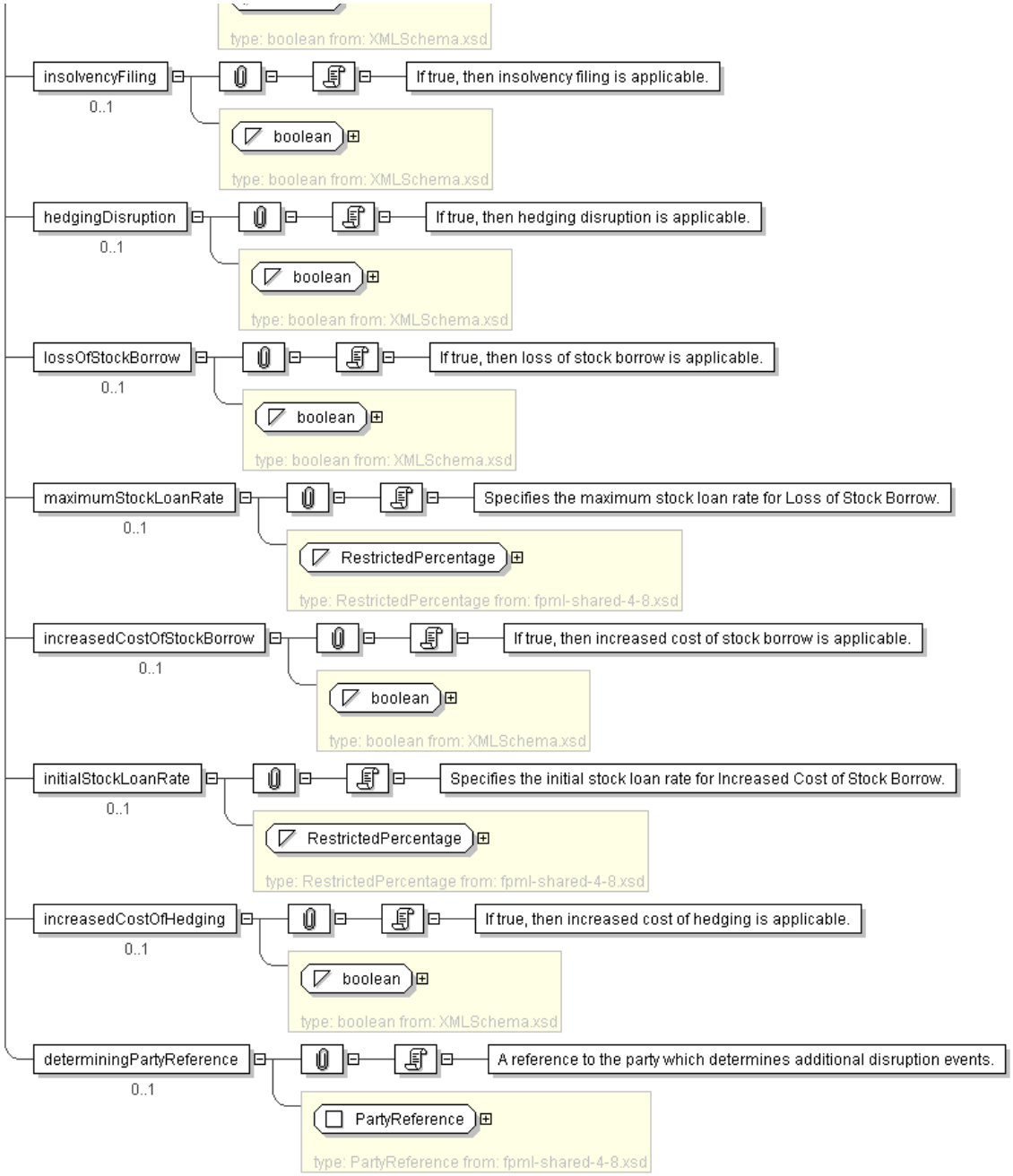
Name	AdditionalDisruptionEvents
Used by (from the same schema document)	Complex Type ExtraordinaryEvents
Abstract	no
Documentation	A type for defining ISDA 2002 Equity Derivative Additional Disruption Events.

XML Instance Representation

```
<...>
  <changeInLaw> xsd:boolean </changeInLaw> [0..1]
  'If true, then change in law is applicable.'

  <failureToDeliver> xsd:boolean </failureToDeliver> [0..1]
  'Where the underlying is shares and the transaction is physically settled, then, if true, a failure to deliver the shares on the settlement
  date will not be an event of default for the purposes of the master agreement.'

  <insolvencyFiling> xsd:boolean </insolvencyFiling> [0..1]
  'If true, then insolvency filing is applicable.'

  <hedgingDisruption> xsd:boolean </hedgingDisruption> [0..1]
  'If true, then hedging disruption is applicable.'

  <lossOfStockBorrow> xsd:boolean </lossOfStockBorrow> [0..1]
  'If true, then loss of stock borrow is applicable.'

  <maximumStockLoanRate> RestrictedPercentage </maximumStockLoanRate> [0..1]
  'Specifies the maximum stock loan rate for Loss of Stock Borrow.'

  <increasedCostOfStockBorrow> xsd:boolean </increasedCostOfStockBorrow> [0..1]
  'If true, then increased cost of stock borrow is applicable.'

  <initialStockLoanRate> RestrictedPercentage </initialStockLoanRate> [0..1]
  'Specifies the initial stock loan rate for Increased Cost of Stock Borrow.'

  <increasedCostOfHedging> xsd:boolean </increasedCostOfHedging> [0..1]
  'If true, then increased cost of hedging is applicable.'

  <determiningPartyReference> PartyReference </determiningPartyReference> [0..1]
  'A reference to the party which determines additional disruption events.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="AdditionalDisruptionEvents">
  <xsd:sequence>
    <xsd:element name="changeInLaw" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="failureToDeliver" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="insolvencyFiling" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="hedgingDisruption" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="lossOfStockBorrow" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="maximumStockLoanRate" type=" RestrictedPercentage " minOccurs="0"/>
```

```
      <xsd:element name="increasedCostOfStockBorrow" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="initialStockLoanRate" type=" RestrictedPercentage " minOccurs="0"/>
      <xsd:element name="increasedCostOfHedging" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="determiningPartyReference" type=" PartyReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdditionalPaymentAmount

[Table of contents]

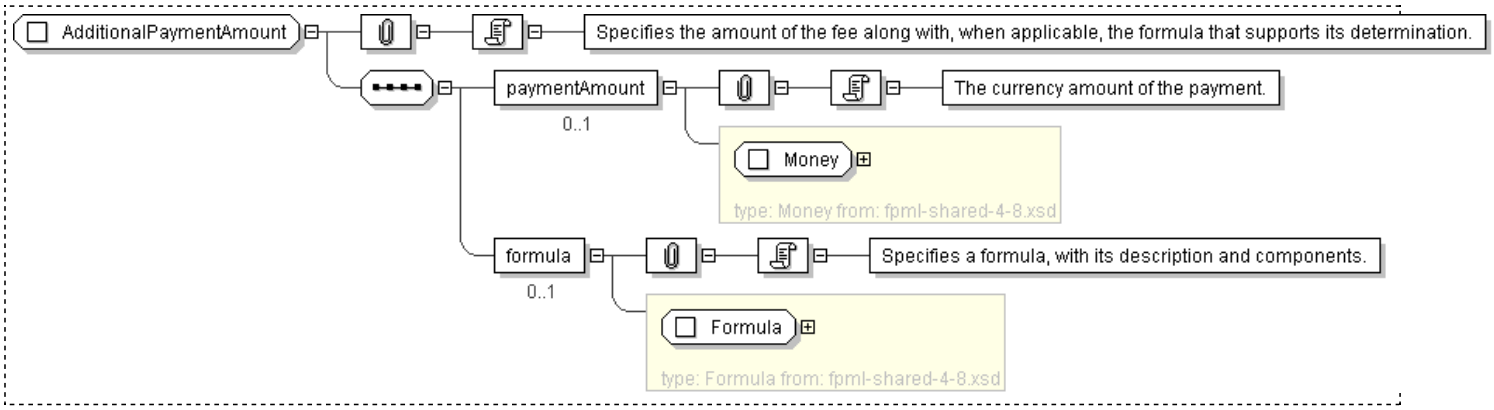
Super-types:	None
Sub-types:	None

Name	AdditionalPaymentAmount
Used by (from the same schema document)	Complex Type ReturnSwapAdditionalPayment
Abstract	no
Documentation	Specifies the amount of the fee along with, when applicable, the formula that supports its determination.

XML Instance Representation

```
<...>
  <paymentAmount> Money </paymentAmount> [0..1]
  'The currency amount of the payment.'

  <formula> Formula </formula> [0..1]
  'Specifies a formula, with its description and components.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdditionalPaymentAmount">
  <xsd:sequence>
    <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
    <xsd:element name="formula" type=" Formula " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableDateOrRelativeDateSequence

[Table of contents]

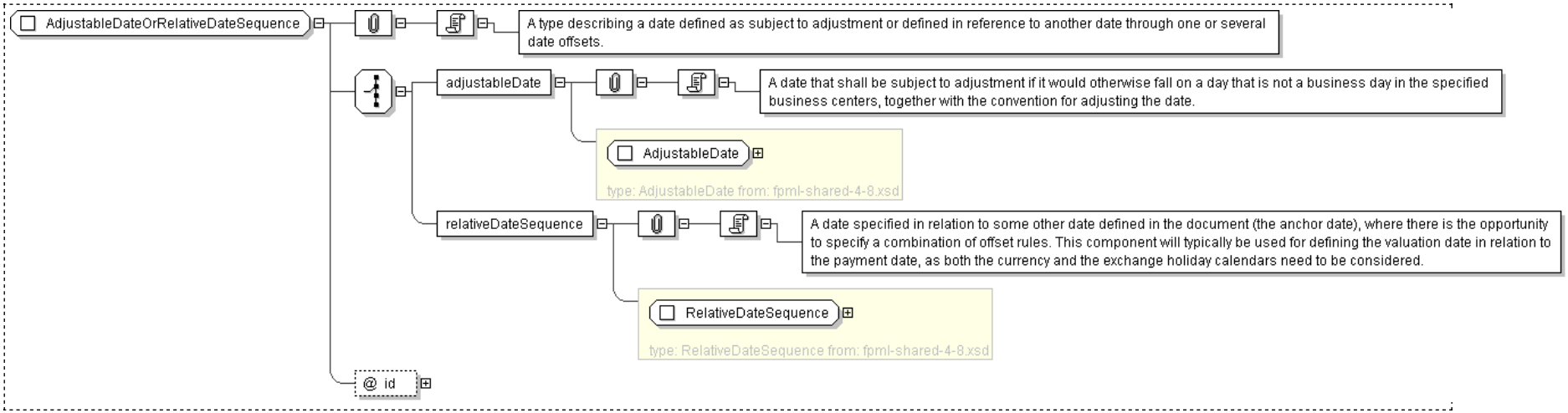
Super-types:	None
Sub-types:	None

Name	AdjustableDateOrRelativeDateSequence
Used by (from the same schema document)	Complex Type EquityValuation
Abstract	no
Documentation	A type describing a date defined as subject to adjustment or defined in reference to another date through one or several date offsets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <adjustableDate> AdjustableDate </adjustableDate> [1]
      'A date that shall be subject to adjustment if it would otherwise fall on a day that is not a business day in the specified business centers,
      together with the convention for adjusting the date.'

      <relativeDateSequence> RelativeDateSequence </relativeDateSequence> [1]
      'A date specified in relation to some other date defined in the document (the anchor date), where there is the opportunity to specify a combination of
      offset rules. This component will typically be used for defining the valuation date in relation to the payment date, as both the currency and the
      exchange holiday calendars need to be considered.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDateOrRelativeDateSequence">
   <xsd:choice>
      <xsd:element name="adjustableDate" type=" AdjustableDate "/>
      <xsd:element name="relativeDateSequence" type=" RelativeDateSequence "/>
   </xsd:choice>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BoundedCorrelation

[Table of contents]

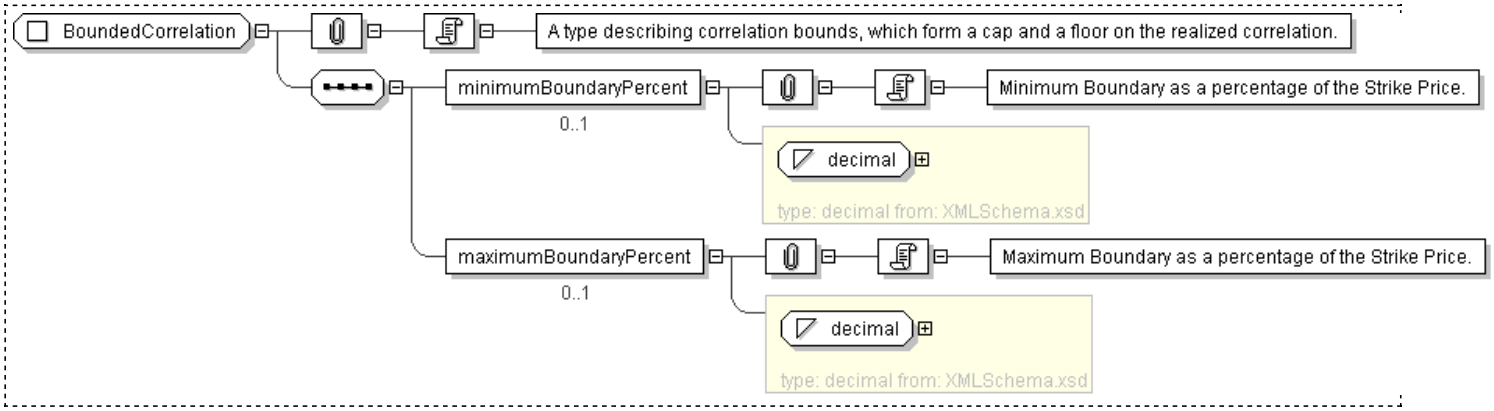
Super-types:	None
Sub-types:	None

Name	BoundedCorrelation
Used by (from the same schema document)	Complex Type Correlation
Abstract	no
Documentation	A type describing correlation bounds, which form a cap and a floor on the realized correlation.

XML Instance Representation

```
<...>
  <minimumBoundaryPercent> xsd:decimal </minimumBoundaryPercent> [0..1]
  'Minimum Boundary as a percentage of the Strike Price.'

  <maximumBoundaryPercent> xsd:decimal </maximumBoundaryPercent> [0..1]
  'Maximum Boundary as a percentage of the Strike Price.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BoundedCorrelation">
  <xsd:sequence>
    <xsd:element name="minimumBoundaryPercent" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="maximumBoundaryPercent" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BoundedVariance

[Table of contents]

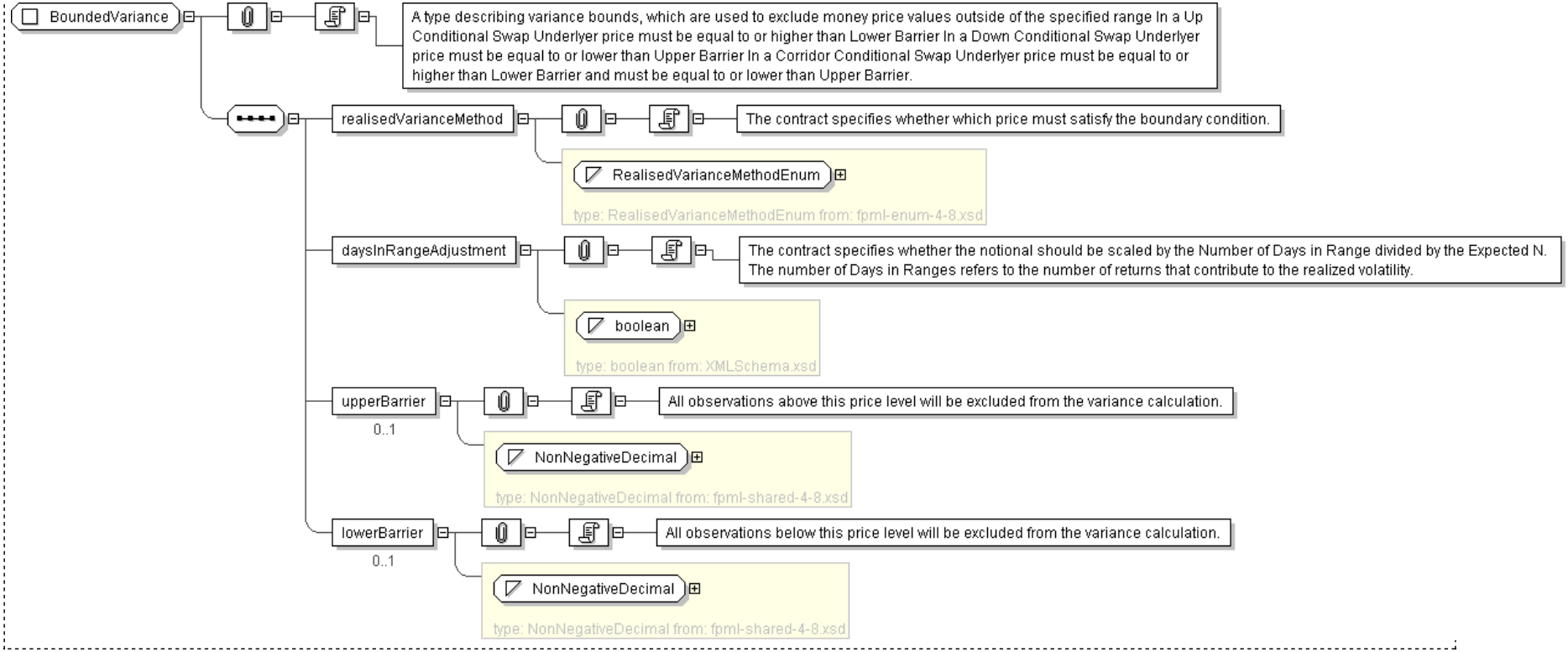
Super-types:	None
Sub-types:	None

Name	BoundedVariance
Used by (from the same schema document)	Complex Type Variance
Abstract	no
Documentation	A type describing variance bounds, which are used to exclude money price values outside of the specified range In a Up Conditional Swap Underlyer price must be equal to or higher than Lower Barrier In a Down Conditional Swap Underlyer price must be equal to or lower than Upper Barrier In a Corridor Conditional Swap Underlyer price must be equal to or higher than Lower Barrier and must be equal to or lower than Upper Barrier.

XML Instance Representation

```
<...>
  <realisedVarianceMethod> RealisedVarianceMethodEnum </realisedVarianceMethod> [1]
  'The contract specifies whether which price must satisfy the boundary condition.'

  <daysInRangeAdjustment> xsd:boolean </daysInRangeAdjustment> [1]
  'The contract specifies whether the notional should be scaled by the Number of Days in Range divided by the Expected N. The number of Days in
  Ranges refers to the number of returns that contribute to the realized volatility.'

  <upperBarrier> NonNegativeDecimal </upperBarrier> [0..1]
  'All observations above this price level will be excluded from the variance calculation.'

  <lowerBarrier> NonNegativeDecimal </lowerBarrier> [0..1]
  'All observations below this price level will be excluded from the variance calculation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BoundedVariance">
  <xsd:sequence>
    <xsd:element name="realisedVarianceMethod" type=" RealisedVarianceMethodEnum "/>
    <xsd:element name="daysInRangeAdjustment" type=" xsd:boolean "/>
    <xsd:element name="upperBarrier" type=" NonNegativeDecimal " minOccurs="0"/>
    <xsd:element name="lowerBarrier" type=" NonNegativeDecimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculatedAmount

[Table of contents]

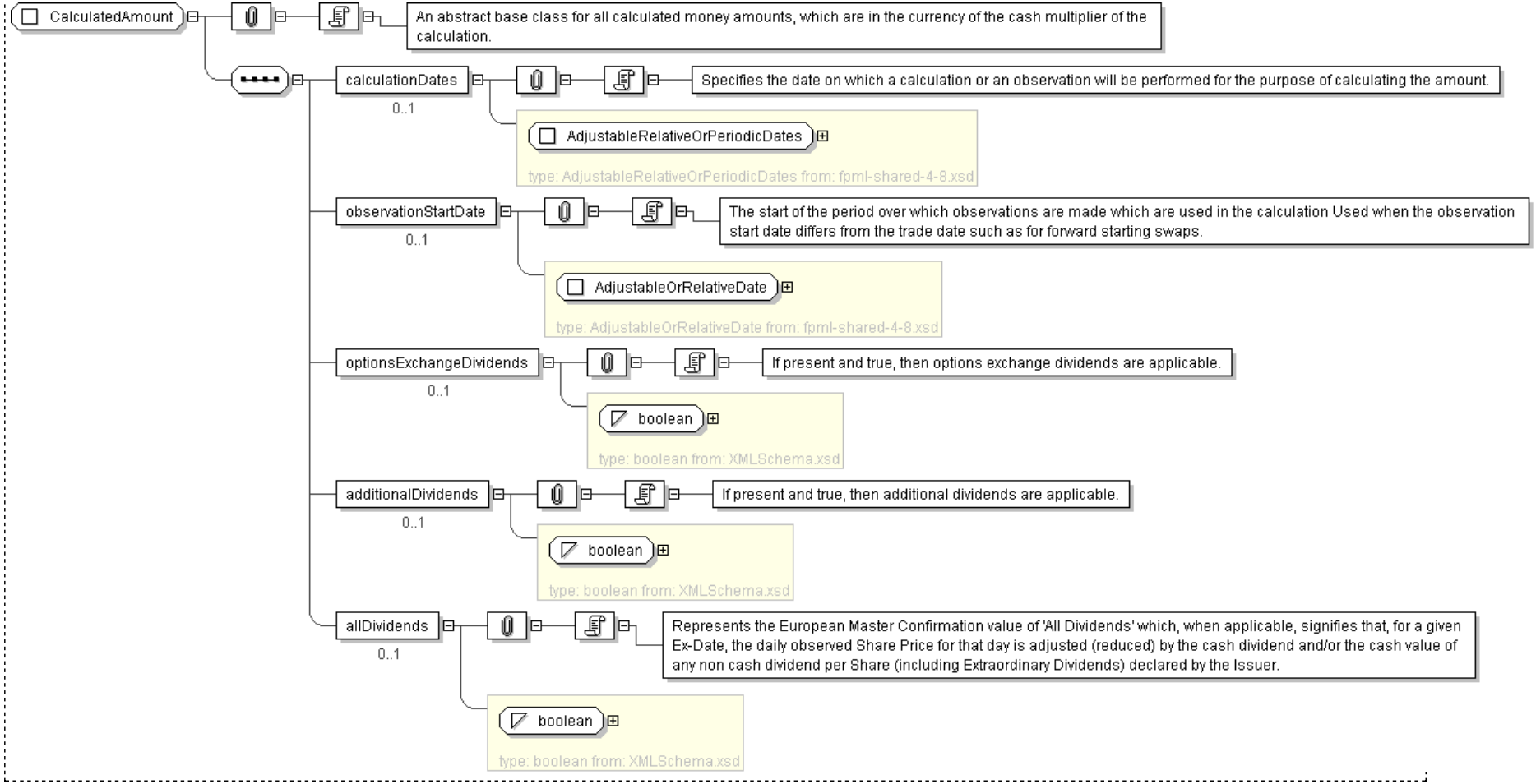
Super-types:	None
Sub-types:	None

Name	CalculatedAmount
Abstract	yes
Documentation	An abstract base class for all calculated money amounts, which are in the currency of the cash multiplier of the calculation.

XML Instance Representation

```
<...>
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for the purpose of calculating the amount.'

   <observationStartDate> AdjustableOrRelativeDate </observationStartDate> [0..1]
   'The start of the period over which observations are made which are used in the calculation Used when the observation start date differs
   from the trade date such as for forward starting swaps.'

   <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
   'If present and true, then options exchange dividends are applicable.'

   <additionalDividends> xsd:boolean </additionalDividends> [0..1]
   'If present and true, then additional dividends are applicable.'

   <allDividends> xsd:boolean </allDividends> [0..1]
   'Represents the European Master Confirmation value of \'All Dividends\' which, when applicable, signifies that, for a given Ex-Date, the
   daily observed Share Price for that day is adjusted (reduced) by the cash dividend and/or the cash value of any non cash dividend per
   Share (including Extraordinary Dividends) declared by the Issuer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculatedAmount" abstract="true">
  <xsd:sequence>
    <xsd:element name="calculationDates" type=" AdjustableRelativeOrPeriodicDates " minOccurs="0"/>
    <xsd:element name="observationStartDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
    <xsd:element name="optionsExchangeDividends" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="additionalDividends" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="allDividends" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationFromObservation

[Table of contents]

Super-types: None

Sub-types:

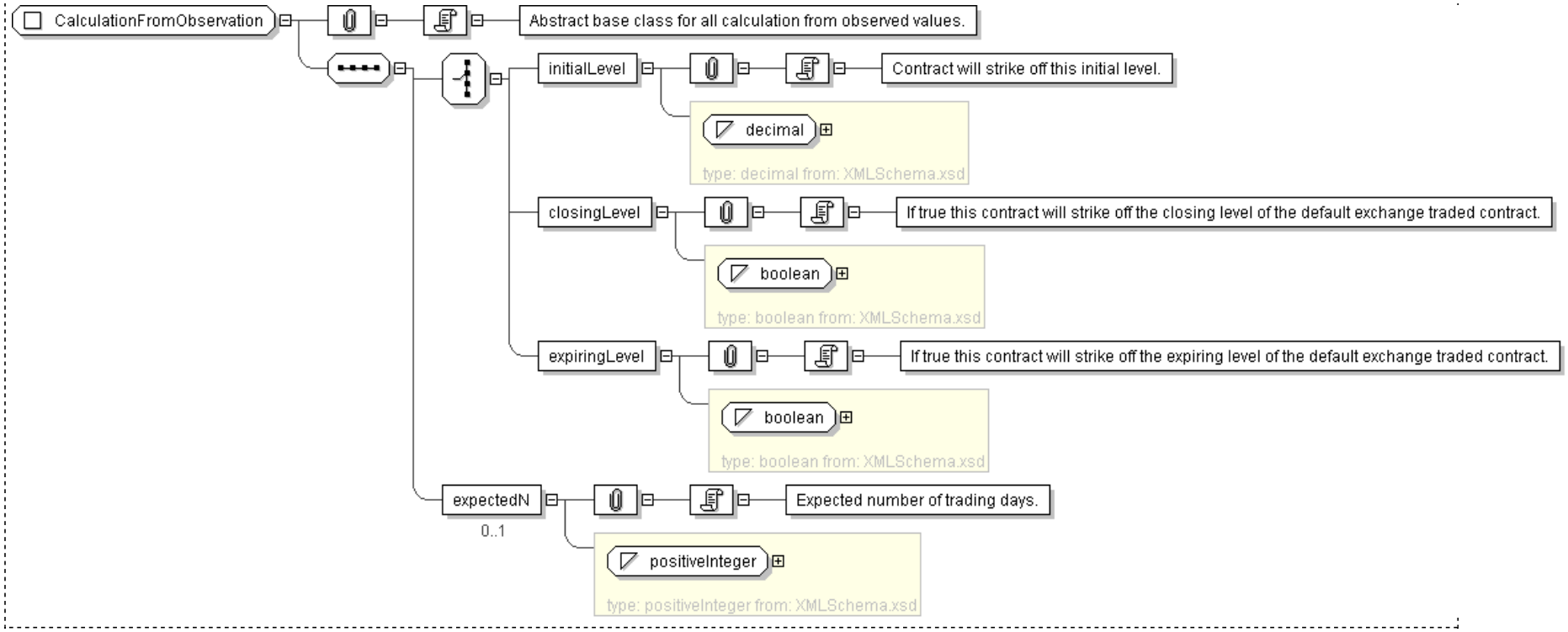
- Correlation (by extension)
- Variance (by extension)

Name	CalculationFromObservation
Abstract	yes
Documentation	Abstract base class for all calculation from observed values.

XML Instance Representation

```
<...>
  Start Choice [1]
    <initialLevel> xsd:decimal </initialLevel> [1]
    'Contract will strike off this initial level.'

    <closingLevel> xsd:boolean </closingLevel> [1]
    'If true this contract will strike off the closing level of the default exchange traded contract.'

    <expiringLevel> xsd:boolean </expiringLevel> [1]
    'If true this contract will strike off the expiring level of the default exchange traded contract.'

  End Choice
  <expectedN> xsd:positiveInteger </expectedN> [0..1]
  'Expected number of trading days.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationFromObservation" abstract="true">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="initialLevel" type=" xsd:decimal "/>
      <xsd:element name="closingLevel" type=" xsd:boolean "/>
      <xsd:element name="expiringLevel" type=" xsd:boolean "/>
    </xsd:choice>
    <xsd:element name="expectedN" type=" xsd:positiveInteger " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Compounding

[Table of contents]

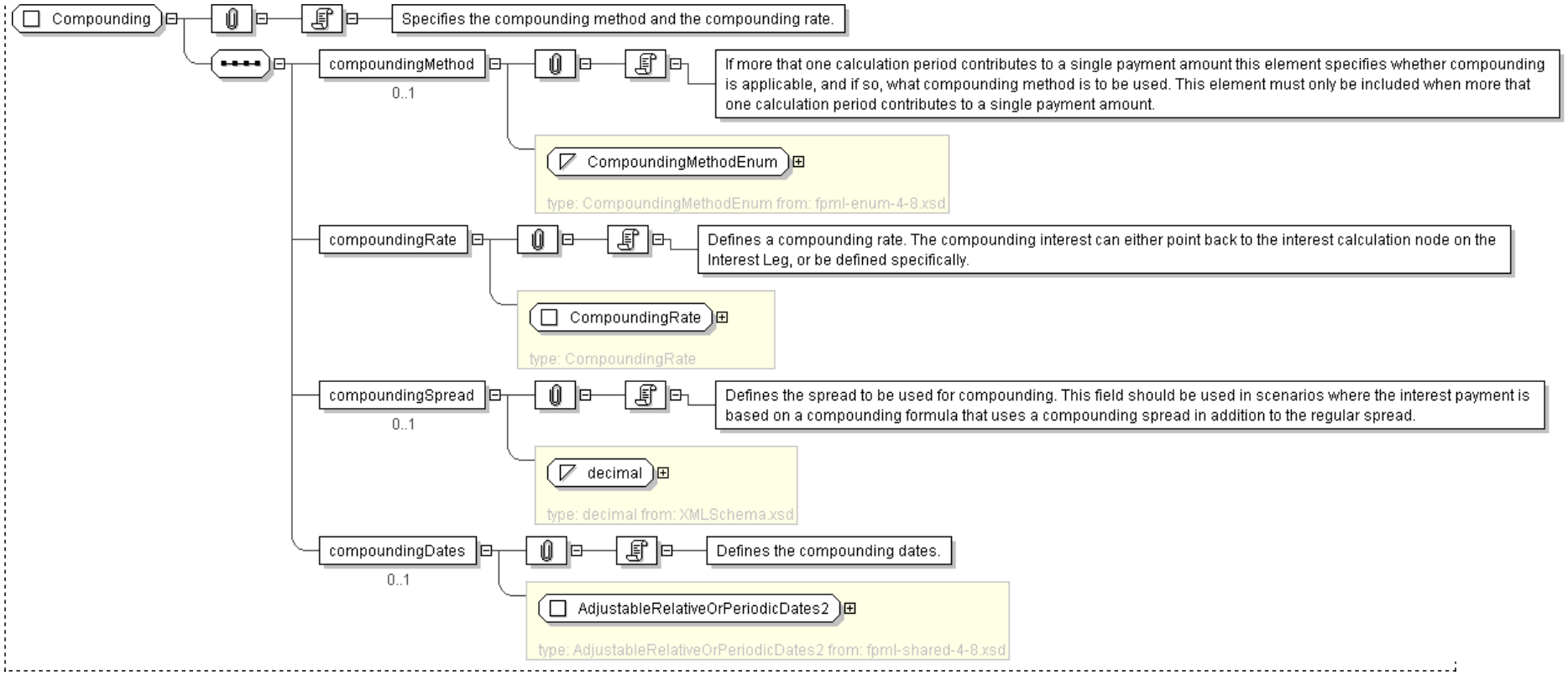
Super-types:	None
Sub-types:	None

Name	Compounding
Used by (from the same schema document)	Complex Type InterestCalculation
Abstract	no
Documentation	Specifies the compounding method and the compounding rate.

XML Instance Representation

```
<...>
   <compoundingMethod> CompoundingMethodEnum </compoundingMethod> [0..1]
   'If more that one calculation period contributes to a single payment amount this element specifies whether compounding is applicable, and
   if so, what compounding method is to be used. This element must only be included when more that one calculation period contributes to a
   single payment amount.'

   <compoundingRate> CompoundingRate </compoundingRate> [1]
   'Defines a compounding rate. The compounding interest can either point back to the interest calculation node on the Interest Leg, or be
   defined specifically.'

   <compoundingSpread> xsd:decimal </compoundingSpread> [0..1]
   'Defines the spread to be used for compounding. This field should be used in scenarios where the interest payment is based on a
   compounding formula that uses a compounding spread in addition to the regular spread.'

   <compoundingDates> AdjustableRelativeOrPeriodicDates2 </compoundingDates> [0..1]
   'Defines the compounding dates.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Compounding">
  <xsd:sequence>
    <xsd:element name="compoundingMethod" type=" CompoundingMethodEnum " minOccurs="0"/>
    <xsd:element name="compoundingRate" type=" CompoundingRate "/>
    <xsd:element name="compoundingSpread" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="compoundingDates" type=" AdjustableRelativeOrPeriodicDates2 " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CompoundingRate

[Table of contents]

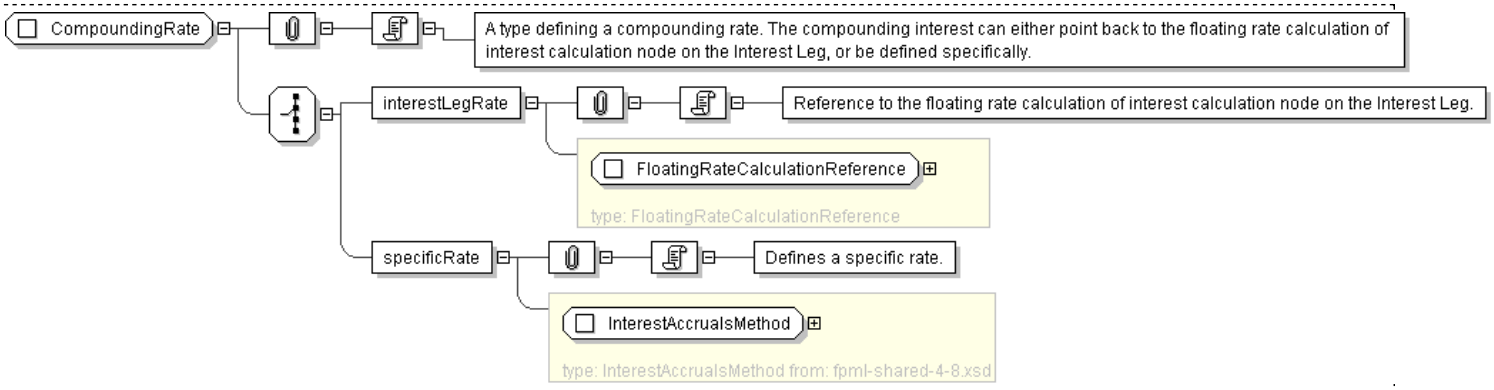
Super-types:	None
Sub-types:	None

Name	CompoundingRate
Used by (from the same schema document)	Complex Type Compounding
Abstract	no
Documentation	A type defining a compounding rate. The compounding interest can either point back to the floating rate calculation of interest calculation node on the Interest Leg, or be defined specifically.

XML Instance Representation

```
<...>
  Start Choice [1]
    <interestLegRate> FloatingRateCalculationReference </interestLegRate> [1]
    'Reference to the floating rate calculation of interest calculation node on the Interest Leg.'

    <specificRate> InterestAccrualsMethod </specificRate> [1]
    'Defines a specific rate.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CompoundingRate">
  <xsd:choice>
    <xsd:element name="interestLegRate" type=" FloatingRateCalculationReference "/>
    <xsd:element name="specificRate" type=" InterestAccrualsMethod "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Correlation

[Table of contents]

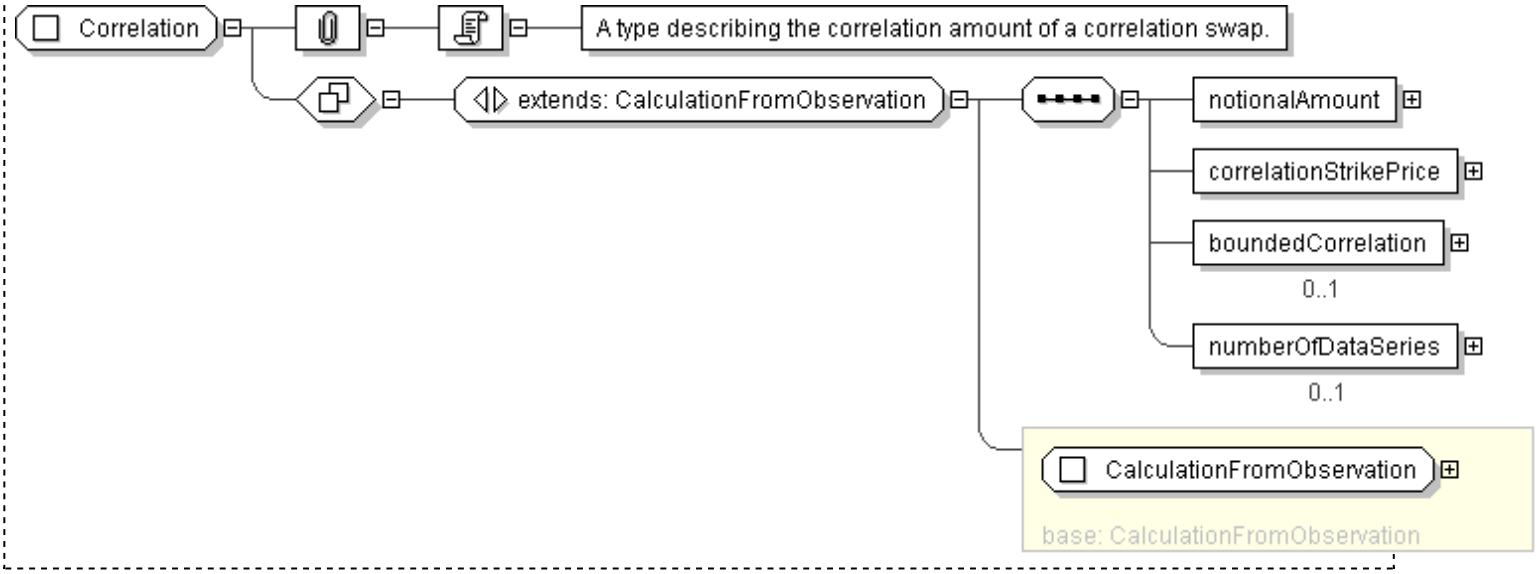
Super-types:	CalculationFromObservation < Correlation (by extension)
Sub-types:	None

Name	Correlation
Abstract	no
Documentation	A type describing the correlation amount of a correlation swap.

XML Instance Representation

```
<...>
   Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      'Contract will strike off this initial level.'

      <closingLevel> xsd:boolean </closingLevel> [1]
      'If true this contract will strike off the closing level of the default
      exchange traded contract.'

      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If true this contract will strike off the expiring level of the default
      exchange traded contract.'

   End Choice
   <expectedN> xsd:positiveInteger </expectedN> [0..1]
   'Expected number of trading days.'

   <notionalAmount> Money </notionalAmount> [1]
   'Notional amount, which is a cash multiplier.'

   <correlationStrikePrice> CorrelationValue </correlationStrikePrice> [1]
   'Correlation Strike Price.'

   <boundedCorrelation> BoundedCorrelation </boundedCorrelation> [0..1]
   'Bounded Correlation.'

   <numberOfDataSeries> xsd:positiveInteger </numberOfDataSeries> [0..1]
   'Number of data series, normal market practice is that correlation data sets are
   drawn from geographic market areas, such as America, Europe and Asia Pacific,
   each of these geographic areas will have its own data series to avoid
   contagion.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Correlation">
   <xsd:complexContent>
      <xsd:extension base=" CalculationFromObservation ">
         <xsd:sequence>
            <xsd:element name="notionalAmount" type=" Money "/>
            <xsd:element name="correlationStrikePrice" type=" CorrelationValue "/>
            <xsd:element name="boundedCorrelation" type=" BoundedCorrelation "
            minOccurs="0"/>
            <xsd:element name="numberOfDataSeries" type=" xsd:positiveInteger "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedVariance

[Table of contents]

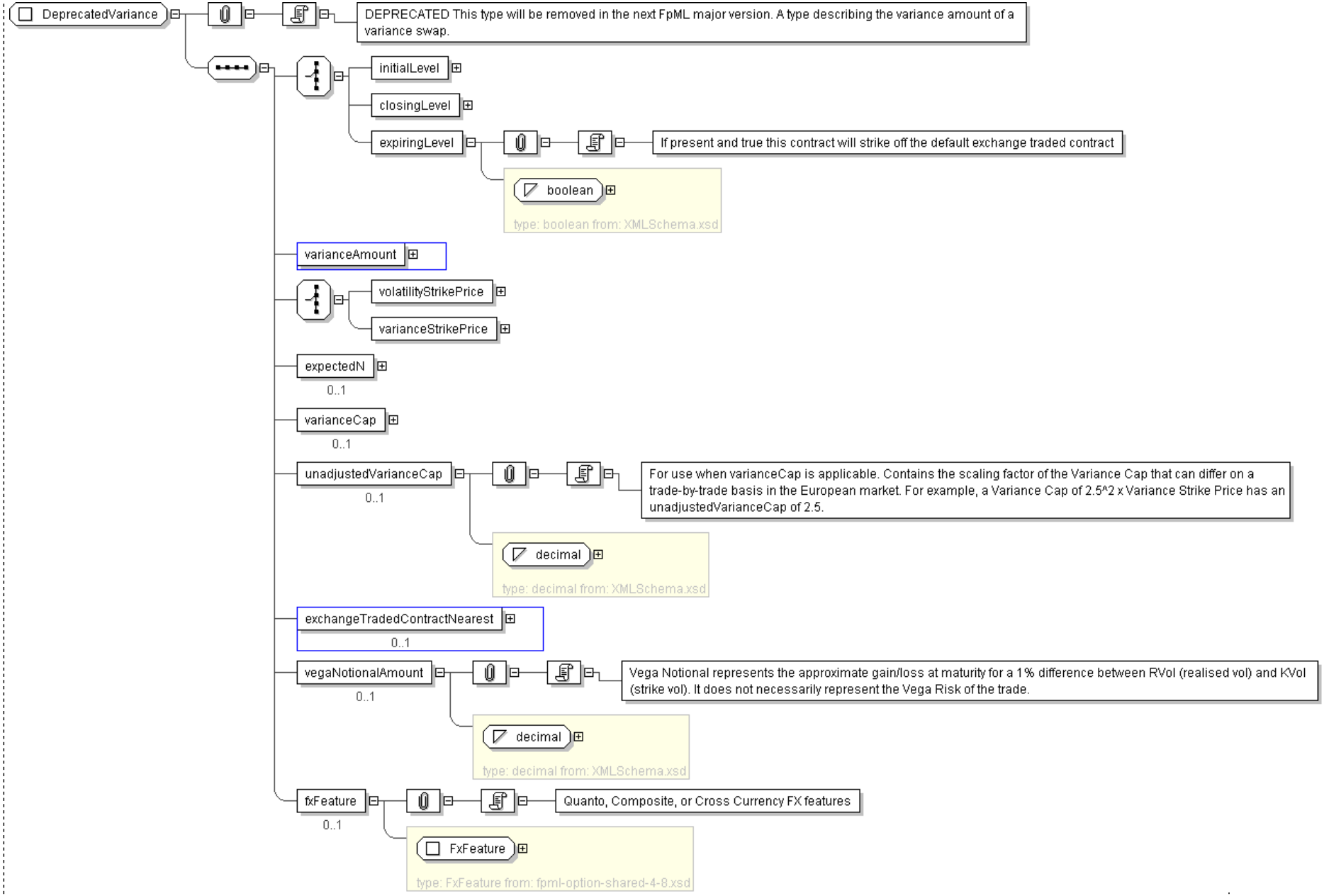
Super-types:	None
Sub-types:	None

Name	DeprecatedVariance
Used by (from the same schema document)	Complex Type LegAmount
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. A type describing the variance amount of a variance swap.

XML Instance Representation

```
<...>
   Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      <closingLevel> xsd:boolean </closingLevel> [1]
      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If present and true this contract will strike off the default exchange traded contract'

   End Choice
   <varianceAmount> Money </varianceAmount> [1]
   Start Choice [1]
      <volatilityStrikePrice> xsd:decimal </volatilityStrikePrice> [1]
      <varianceStrikePrice> xsd:decimal </varianceStrikePrice> [1]
   End Choice
   <expectedN> xsd:integer </expectedN> [0..1]
   <varianceCap> xsd:boolean </varianceCap> [0..1]
   <unadjustedVarianceCap> xsd:decimal </unadjustedVarianceCap> [0..1]
   'For use when varianceCap is applicable. Contains the scaling factor of the Variance Cap that can differ on a trade-by-trade basis in the
   European market. For example, a Variance Cap of 2.5^2 x Variance Strike Price has an unadjustedVarianceCap of 2.5.'

   <exchangeTradedContractNearest> ExchangeTradedContract </exchangeTradedContractNearest> [0..1]
   <vegaNotionalAmount> xsd:decimal </vegaNotionalAmount> [0..1]
   'Vega Notional represents the approximate gain/loss at maturity for a 1% difference between RVol (realised vol) and KVol (strike vol). It
   does not necessarily represent the Vega Risk of the trade.'

   <fxFeature> FxFeature </fxFeature> [0..1]
   'Quanto, Composite, or Cross Currency FX features'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVariance" deprecated="true" deprecatedReason="Use new Variance complex type">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="initialLevel" type=" xsd:decimal "/>
      <xsd:element name="closingLevel" type=" xsd:boolean "/>
      <xsd:element name="expiringLevel" type=" xsd:boolean "/>
    </xsd:choice>
    <xsd:element name="varianceAmount" type=" Money "/>
    <xsd:choice>
      <xsd:element name="volatilityStrikePrice" type=" xsd:decimal "/>
      <xsd:element name="varianceStrikePrice" type=" xsd:decimal "/>
```

```
      </xsd:choice>
      <xsd:element name="expectedN" type=" xsd:integer " minOccurs="0"/>
      <xsd:element name="varianceCap" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="unadjustedVarianceCap" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract " minOccurs="0"/>
      <xsd:element name="vegaNotionalAmount" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedVarianceAmount

[Table of contents]

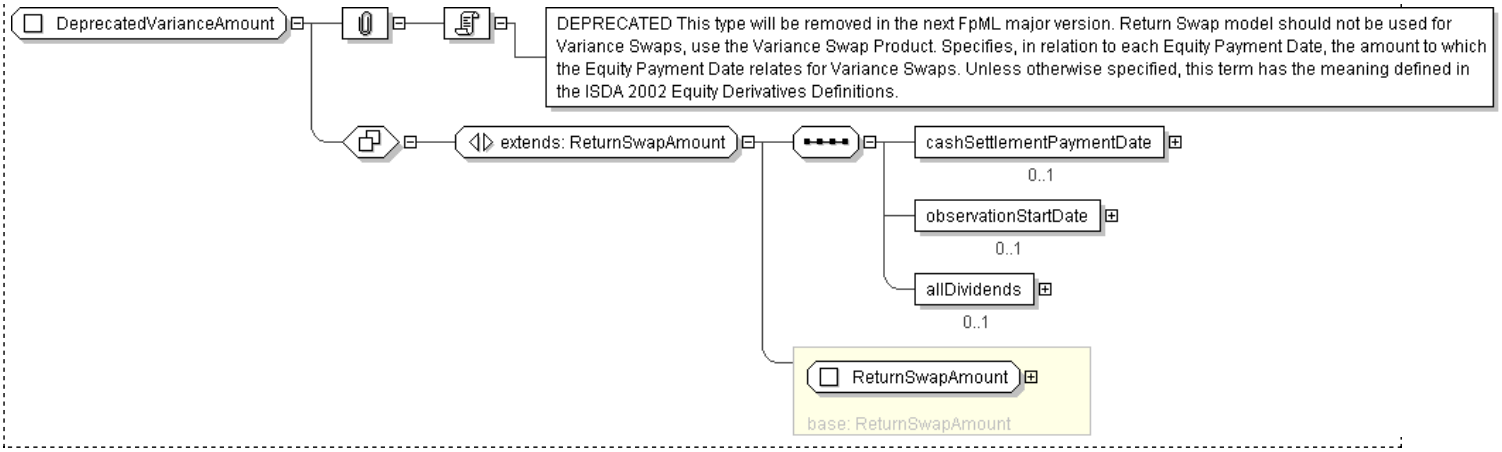
Super-types:	LegAmount < ReturnSwapAmount (by extension) < DeprecatedVarianceAmount (by extension)
Sub-types:	None

Name	DeprecatedVarianceAmount
Used by (from the same schema document)	Complex Type DeprecatedVarianceLeg
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates for Variance Swaps. Unless otherwise specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.

XML Instance Representation

```
<...>
  Start Group: CurrencyAndDeterminationMethod.model [0..1]
    Start Choice [1]
      <currency> Currency </currency> [1]
        'The currency in which an amount is denominated.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
        'Specifies the method according to which an amount or a date is determined.'

      <currencyReference> IdentifiedCurrencyReference </currencyReference> [1]
        'Reference to a currency defined elsewhere in the document'

    End Choice
  End Group: CurrencyAndDeterminationMethod.model
  <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
    'DEPRECATED. Currency in which the payment relating to the leg amount (equity amount or interest amount) or the
    dividend will be denominated.'

  Start Choice [1]
    <referenceAmount> ReferenceAmount </referenceAmount> [1]
      'Specifies the reference Amount when this term either corresponds to the standard ISDA Definition (either the 2002
      Equity Definition for the Equity Amount, or the 2000 Definition for the Interest Amount), or points to a term
      defined elsewhere in the swap document.'

    <formula> Formula </formula> [1]
      'Specifies a formula, with its description and components.'

    <encodedDescription> xsd:base64Binary </encodedDescription> [1]
      'Description of the leg amount when represented through an encoded image.'

    <variance> DeprecatedVariance </variance> [1]
      'DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for
      Variance Swaps, use the Variance Swap Product. Specifies Variance for Variance Leg.'

  End Choice
  <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
    'Specifies the date on which a calculation or an observation will be performed for the purpose of defining the
    Equity Amount, and in accordance to the definition terms of this latter.'

  <cashSettlement> xsd:boolean </cashSettlement> [1]
    'If true, then cash settlement is applicable.'

  <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
    'If present and true, then options exchange dividends are applicable.'

  <additionalDividends> xsd:boolean </additionalDividends> [0..1]
    'If present and true, then additional dividends are applicable.'

  <cashSettlementPaymentDate> AdjustableOrRelativeDate </cashSettlementPaymentDate> [0..1]
    'Typically specified as a number of days following the valuation date, such as one settlement cycle following the
    valuation date. Number of days can vary in the European market.'

  <observationStartDate> StartingDate </observationStartDate> [0..1]
    'The start of the period over which observations are made to determine the variance. Used when the date differs from
    the trade date such as for forward starting variance swaps.'

  <allDividends> xsd:boolean </allDividends> [0..1]
    'Represents the European Master Confirmation value of \"All Dividends\" which, when applicable, signifies that, for a
    given Ex-Date, the daily observed Share Price for that day is adjusted (reduced) by the cash dividend and/or the
    cash value of any non cash dividend per Share (including Extraordinary Dividends) declared by the Issuer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVarianceAmount" deprecated="true" deprecatedReason="Return Swap model should not be used
for Variance Swaps, use the Variance Swap Product">
  <xsd:complexContent>
    <xsd:extension base=" ReturnSwapAmount ">
      <xsd:sequence>
        <xsd:element name="cashSettlementPaymentDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
        <xsd:element name="observationStartDate" type=" StartingDate " minOccurs="0"/>
        <xsd:element name="allDividends" type=" xsd:boolean " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedVarianceLeg

[Table of contents]

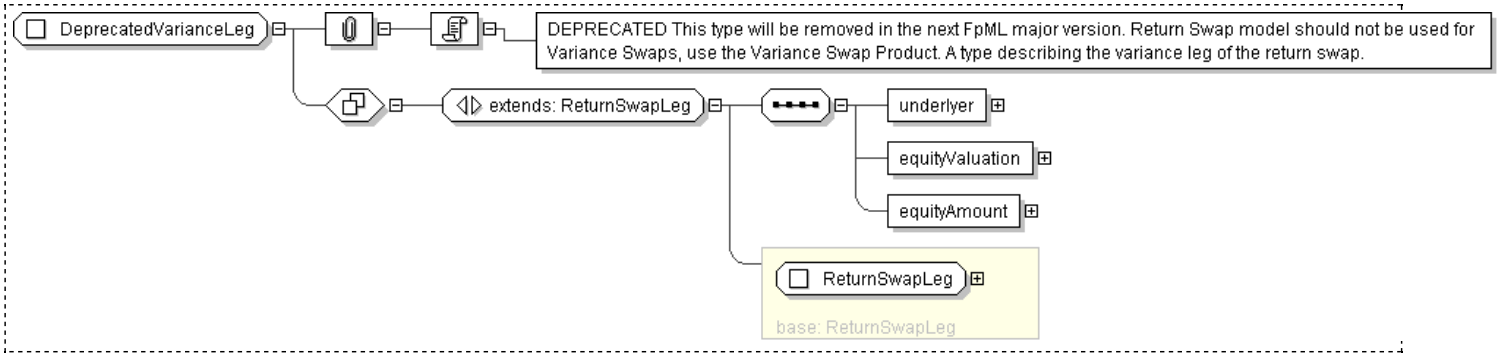
Super-types:	Leg < ReturnSwapLeg (by extension) < DeprecatedVarianceLeg (by extension)
Sub-types:	None

Name	DeprecatedVarianceLeg
Used by (from the same schema document)	Element varianceLeg
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. A type describing the variance leg of the return swap.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <equityValuation> EquityValuation </equityValuation> [1]
   'Valuation of the underlyer.'

   <equityAmount> DeprecatedVarianceAmount </equityAmount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates. Unless
   otherwise specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVarianceLeg" deprecated="true" deprecatedReason="Return Swap model should not be used
for Variance Swaps, use the Variance Swap Product">
  <xsd:complexContent>
    <xsd:extension base=" ReturnSwapLeg ">
      <xsd:sequence>
        <xsd:element name="underlyer" type=" Underlyer "/>
        <xsd:element name="equityValuation" type=" EquityValuation "/>
        <xsd:element name="equityAmount" type=" DeprecatedVarianceAmount "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DirectionalLeg

[Table of contents]

Super-types: Leg < DirectionalLeg (by extension)

Sub-types:

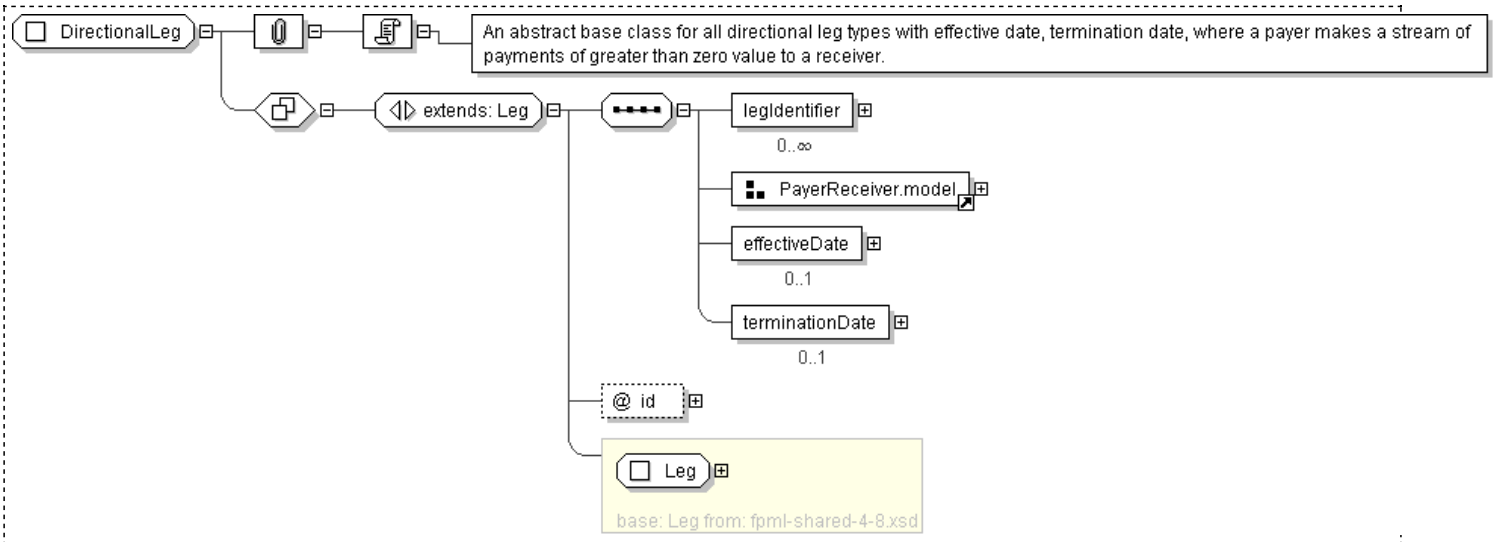
- DirectionalLegUnderlyer (by extension)
  - DirectionalLegUnderlyerValuation (by extension)

Name	DirectionalLeg
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
   'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a date specified
   somewhere else in the document (through the relativeDate component), this element will typically point to
   the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
   'Specifies the termination date of this leg of the swap. When defined in relation to a date specified
   somewhere else in the document (through the relativeDate component), this element will typically point to
   the termination date of the other leg of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLeg" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:element name="legIdentifier" type=" LegIdentifier " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
        <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DirectionalLegUnderlyer

[Table of contents]

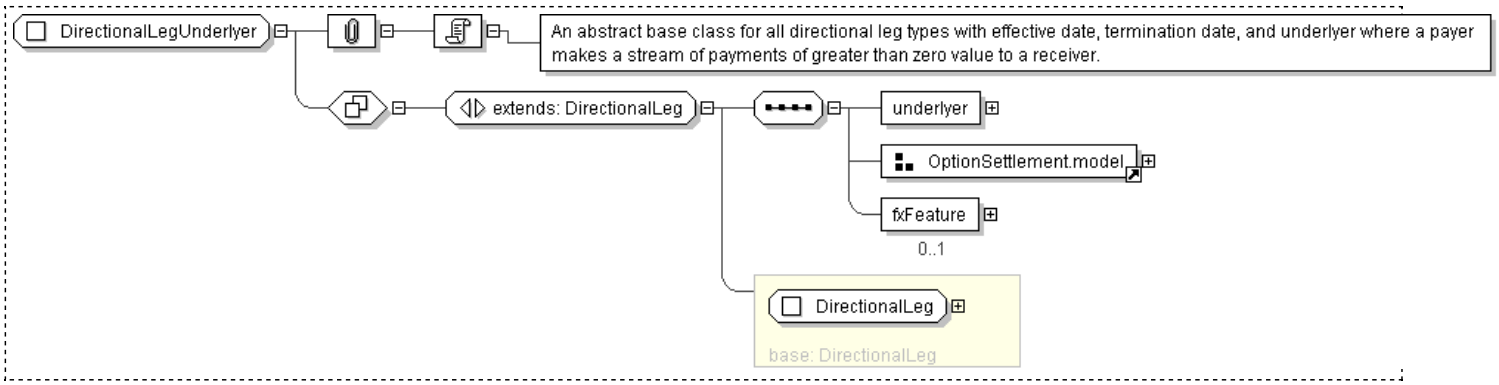
Super-types:	Leg < DirectionalLeg (by extension) < DirectionalLegUnderlyer (by extension)
Sub-types:	• DirectionalLegUnderlyerValuation (by extension)

Name	DirectionalLegUnderlyer
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, and underlyer where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
   'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere
   else in the document (through the relativeDate component), this element will typically point to the effective
   date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
   'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere
   else in the document (through the relativeDate component), this element will typically point to the termination
   date of the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
         'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
         'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <fxFeature> FxFeature </fxFeature> [0..1]
   'Quanto, Composite, or Cross Currency FX features.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLegUnderlyer" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" DirectionalLeg ">
         <xsd:sequence>
            <xsd:element name="underlyer" type=" Underlyer "/>
            <xsd:group ref=" OptionSettlement.model "/>
            <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
         </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DirectionalLegUnderlyerValuation

[Table of contents]

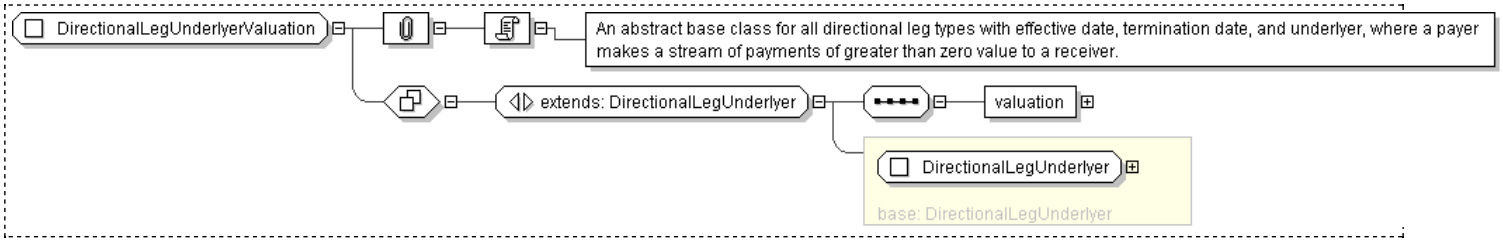
Super-types:	Leg < DirectionalLeg (by extension) < DirectionalLegUnderlyer (by extension) < DirectionalLegUnderlyerValuation (by extension)
Sub-types:	None

Name	DirectionalLegUnderlyerValuation
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, and underlyer, where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
    'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
    'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere else in
    the document (through the relativeDate component), this element will typically point to the effective date of the
    other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
    'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere else
    in the document (through the relativeDate component), this element will typically point to the termination date of
    the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlyer of the leg.'

   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
          'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
          'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <fxFeature> FxFeature </fxFeature> [0..1]
    'Quanto, Composite, or Cross Currency FX features.'

   <valuation> EquityValuation </valuation> [1]
    'Valuation of the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLegUnderlyerValuation" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" DirectionalLegUnderlyer ">
      <xsd:sequence>
        <xsd:element name="valuation" type=" EquityValuation "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendAdjustment

[Table of contents]

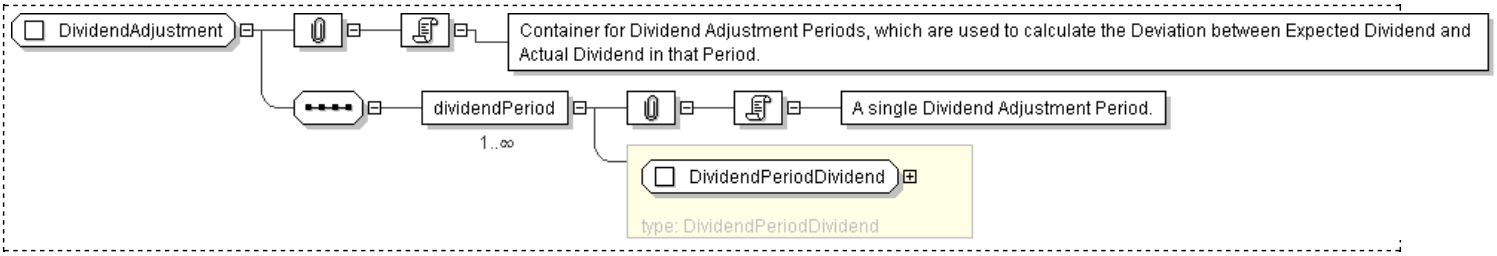
Super-types:	None
Sub-types:	None

Name	DividendAdjustment
Used by (from the same schema document)	Complex Type OptionFeatures
Abstract	no
Documentation	Container for Dividend Adjustment Periods, which are used to calculate the Deviation between Expected Dividend and Actual Dividend in that Period.

XML Instance Representation

```
<...>
  <dividendPeriod> DividendPeriodDividend </dividendPeriod> [1..*]
  'A single Dividend Adjustment Period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendAdjustment">
  <xsd:sequence>
    <xsd:element name="dividendPeriod" type=" DividendPeriodDividend " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendConditions

[Table of contents]

Super-types:	None
Sub-types:	None

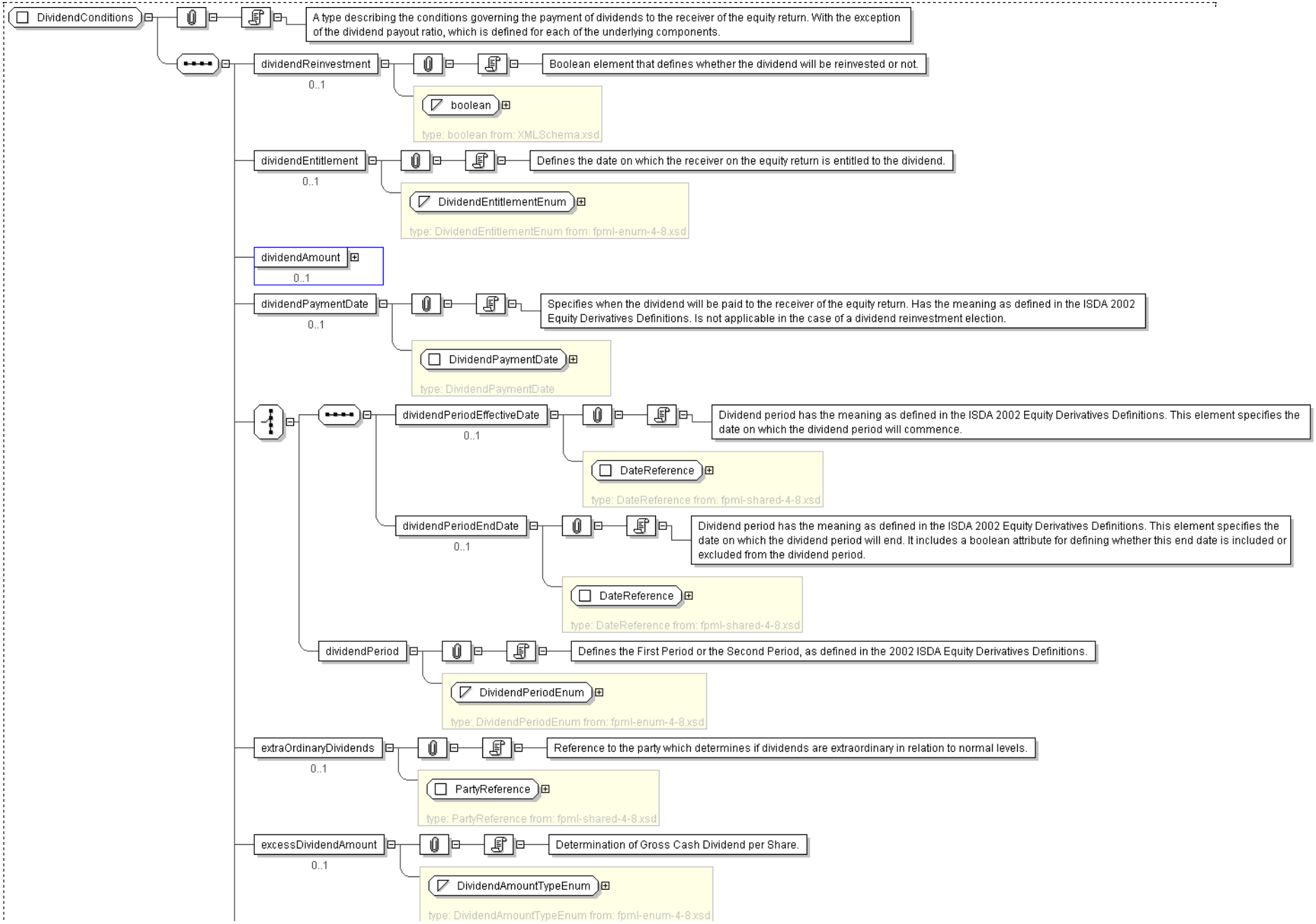
Name	DividendConditions
Used by (from the same schema document)	Complex Type Return
Abstract	no
Documentation	A type describing the conditions governing the payment of dividends to the receiver of the equity return. With the exception of the dividend payout ratio, which is defined for each of the underlying components.

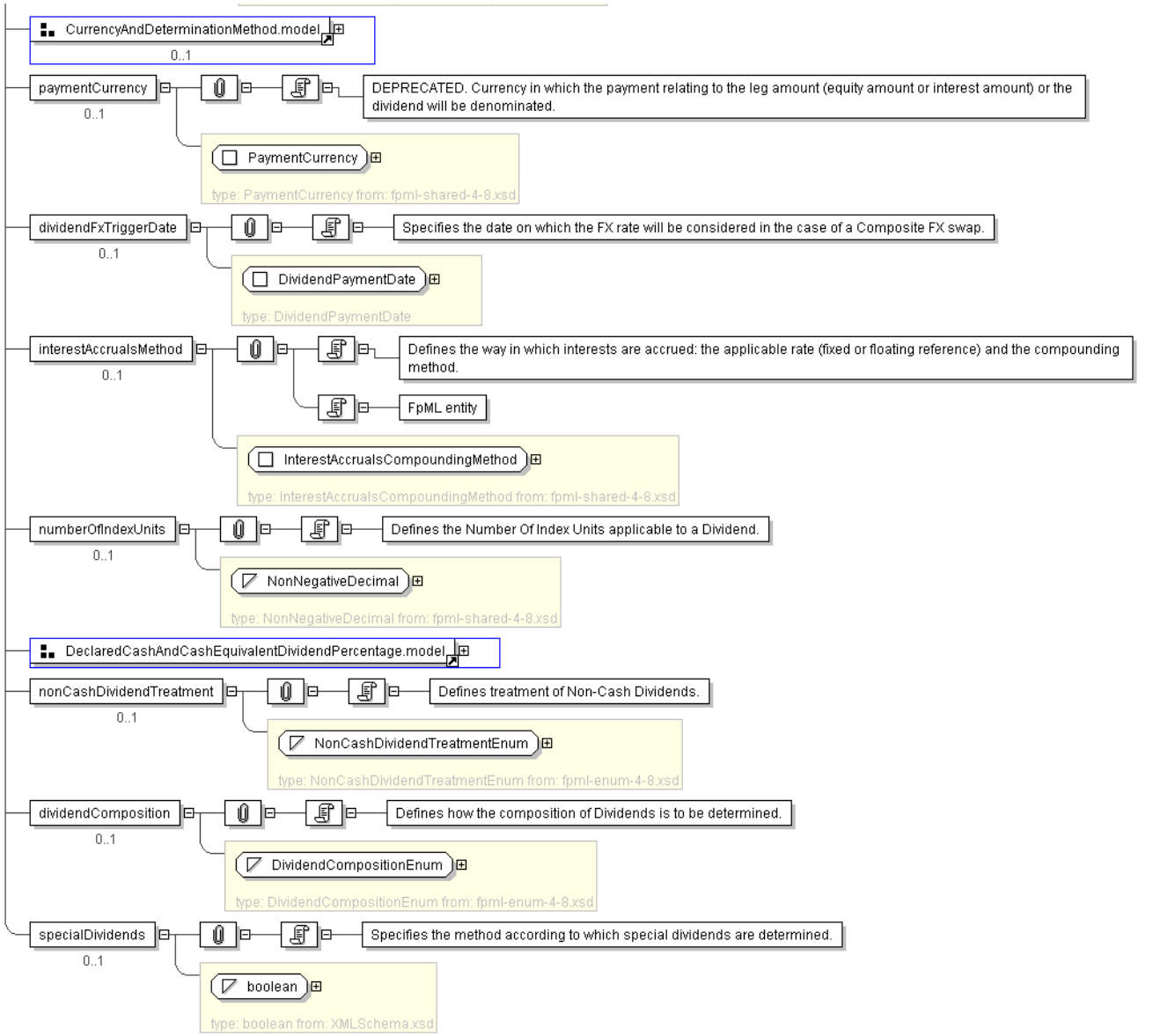
XML Instance Representation

```
<...>
  <dividendReinvestment> xsd:boolean </dividendReinvestment> [0..1]
  'Boolean element that defines whether the dividend will be reinvested or not.'

  <dividendEntitlement> DividendEntitlementEnum </dividendEntitlement> [0..1]
  'Defines the date on which the receiver on the equity return is entitled to the dividend.'

  <dividendAmount> DividendAmountTypeEnum </dividendAmount> [0..1]
  <dividendPaymentDate> DividendPaymentDate </dividendPaymentDate> [0..1]
  'Specifies when the dividend will be paid to the receiver of the equity return. Has the meaning as defined in the ISDA 2002 Equity Derivatives Definitions. Is not
  applicable in the case of a dividend reinvestment election.'

  Start Choice [1]
    <dividendPeriodEffectiveDate> DateReference </dividendPeriodEffectiveDate> [0..1]
    'Dividend period has the meaning as defined in the ISDA 2002 Equity Derivatives Definitions. This element specifies the date on which the dividend period will
    commence.'

    <dividendPeriodEndDate> DateReference </dividendPeriodEndDate> [0..1]
    'Dividend period has the meaning as defined in the ISDA 2002 Equity Derivatives Definitions. This element specifies the date on which the dividend period will
    end. It includes a boolean attribute for defining whether this end date is included or excluded from the dividend period.'

    <dividendPeriod> DividendPeriodEnum </dividendPeriod> [1]
    'Defines the First Period or the Second Period, as defined in the 2002 ISDA Equity Derivatives Definitions.'

  End Choice
  <extraOrdinaryDividends> PartyReference </extraOrdinaryDividends> [0..1]
  'Reference to the party which determines if dividends are extraordinary in relation to normal levels.'

  <excessDividendAmount> DividendAmountTypeEnum </excessDividendAmount> [0..1]
  'Determination of Gross Cash Dividend per Share.'

  Start Group: CurrencyAndDeterminationMethod.model [0..1]
    Start Choice [1]
      <currency> Currency </currency> [1]
      'The currency in which an amount is denominated.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <currencyReference> IdentifiedCurrencyReference </currencyReference> [1]
      'Reference to a currency defined elsewhere in the document'

    End Choice
  End Group: CurrencyAndDeterminationMethod.model
  <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
  'DEPRECATED. Currency in which the payment relating to the leg amount (equity amount or interest amount) or the dividend will be denominated.'

  <dividendFxTriggerDate> DividendPaymentDate </dividendFxTriggerDate> [0..1]
  'Specifies the date on which the FX rate will be considered in the case of a Composite FX swap.'

  <interestAccrualsMethod> InterestAccrualsCompoundingMethod </interestAccrualsMethod> [0..1]
  'Defines the way in which interests are accrued: the applicable rate (fixed or floating reference) and the compounding method.','FpML entity'

  <numberOfIndexUnits> NonNegativeDecimal </numberOfIndexUnits> [0..1]
  'Defines the Number Of Index Units applicable to a Dividend.'

  <declaredCashDividendPercentage> NonNegativeDecimal </declaredCashDividendPercentage> [0..1]
  'Declared Cash Dividend Percentage.'

  <declaredCashEquivalentDividendPercentage> NonNegativeDecimal </declaredCashEquivalentDividendPercentage> [0..1]
  'Declared Cash Equivalent Dividend Percentage.'
```

```
<nonCashDividendTreatment> NonCashDividendTreatmentEnum </nonCashDividendTreatment> [0..1]
'Defines treatment of Non-Cash Dividends.'

<dividendComposition> DividendCompositionEnum </dividendComposition> [0..1]
'Defines how the composition of Dividends is to be determined.'

<specialDividends> xsd:boolean </specialDividends> [0..1]
'Specifies the method according to which special dividends are determined.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="DividendConditions">
  <xsd:sequence>
    <xsd:element name="dividendReinvestment" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="dividendEntitlement" type=" DividendEntitlementEnum " minOccurs="0"/>
    <xsd:element name="dividendAmount" type=" DividendAmountTypeEnum " minOccurs="0"/>
    <xsd:element name="dividendPaymentDate" type=" DividendPaymentDate " minOccurs="0"/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="dividendPeriodEffectiveDate" type=" DateReference " minOccurs="0"/>
        <xsd:element name="dividendPeriodEndDate" type=" DateReference " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="dividendPeriod" type=" DividendPeriodEnum "/>
    </xsd:choice>
    <xsd:element name="extraOrdinaryDividends" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="excessDividendAmount" type=" DividendAmountTypeEnum " minOccurs="0"/>
    <xsd:group ref=" CurrencyAndDeterminationMethod.model " minOccurs="0"/>
    <xsd:element name="paymentCurrency" type=" PaymentCurrency " minOccurs="0" deprecated="true" deprecatedReason="The model is wrong since it has an intradocument
    reference that is not clear. Current PaymentCurrency model and elements using this type are deprecated. Instead, ReturnSwapCurrency.model above, the choice between
```

```
      currency, determinationMethod, and currencyReference (of type CurrencyReference) ,should be used."/>
      <xsd:element name="dividendFxTriggerDate" type=" DividendPaymentDate " minOccurs="0"/>
      <xsd:element name="interestAccrualsMethod" type=" InterestAccrualsCompoundingMethod " minOccurs="0"/>
      <xsd:element name="numberOfIndexUnits" type=" NonNegativeDecimal " minOccurs="0"/>
      <xsd:group ref=" DeclaredCashAndCashEquivalentDividendPercentage.model "/>
      <xsd:element name="nonCashDividendTreatment" type=" NonCashDividendTreatmentEnum " minOccurs="0"/>
      <xsd:element name="dividendComposition" type=" DividendCompositionEnum " minOccurs="0"/>
      <xsd:element name="specialDividends" type=" xsd:boolean " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendPaymentDate

[Table of contents]

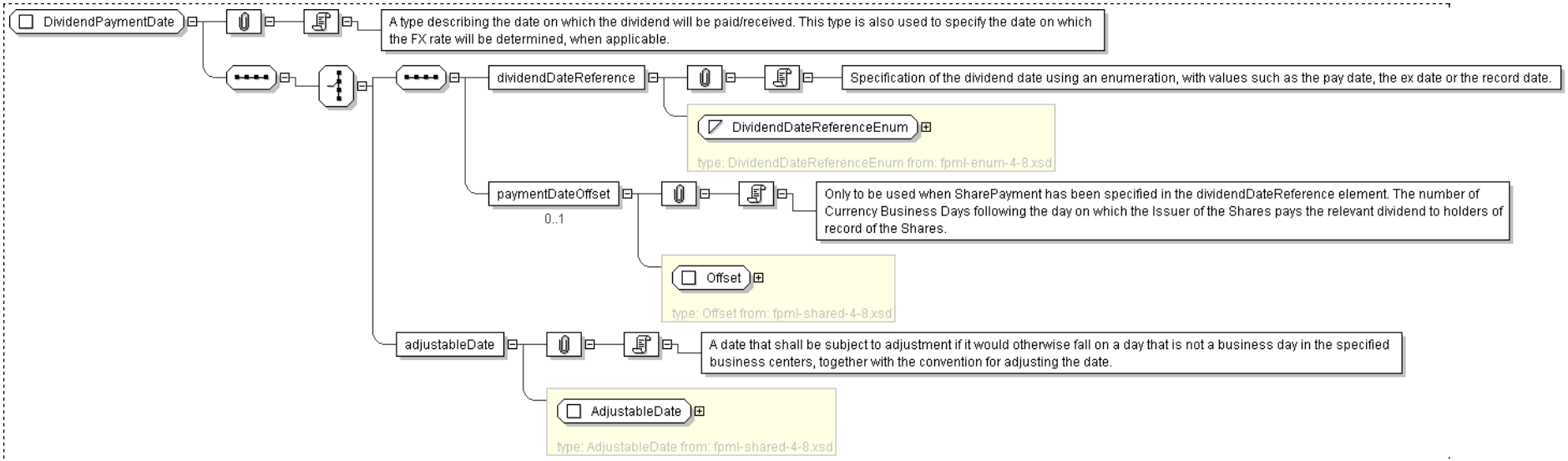
Super-types:	None
Sub-types:	None

Name	DividendPaymentDate
Used by (from the same schema document)	Complex Type DividendConditions , Complex Type DividendConditions
Abstract	no
Documentation	A type describing the date on which the dividend will be paid/received. This type is also used to specify the date on which the FX rate will be determined, when applicable.

XML Instance Representation

```
<...>
  Start Choice [1]
    <dividendDateReference> DividendDateReferenceEnum </dividendDateReference> [1]
    'Specification of the dividend date using an enumeration, with values such as the pay date, the ex date or the record date.'

    <paymentDateOffset> Offset </paymentDateOffset> [0..1]
    'Only to be used when SharePayment has been specified in the dividendDateReference element. The number of Currency Business Days following the day on which the
    Issuer of the Shares pays the relevant dividend to holders of record of the Shares.'

    <adjustableDate> AdjustableDate </adjustableDate> [1]
    'A date that shall be subject to adjustment if it would otherwise fall on a day that is not a business day in the specified business centers, together with
    the convention for adjusting the date.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPaymentDate">
  <xsd:sequence>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="dividendDateReference" type=" DividendDateReferenceEnum "/>
        <xsd:element name="paymentDateOffset" type=" Offset " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="adjustableDate" type=" AdjustableDate "/>
    </xsd:choice>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendPeriod

[Table of contents]

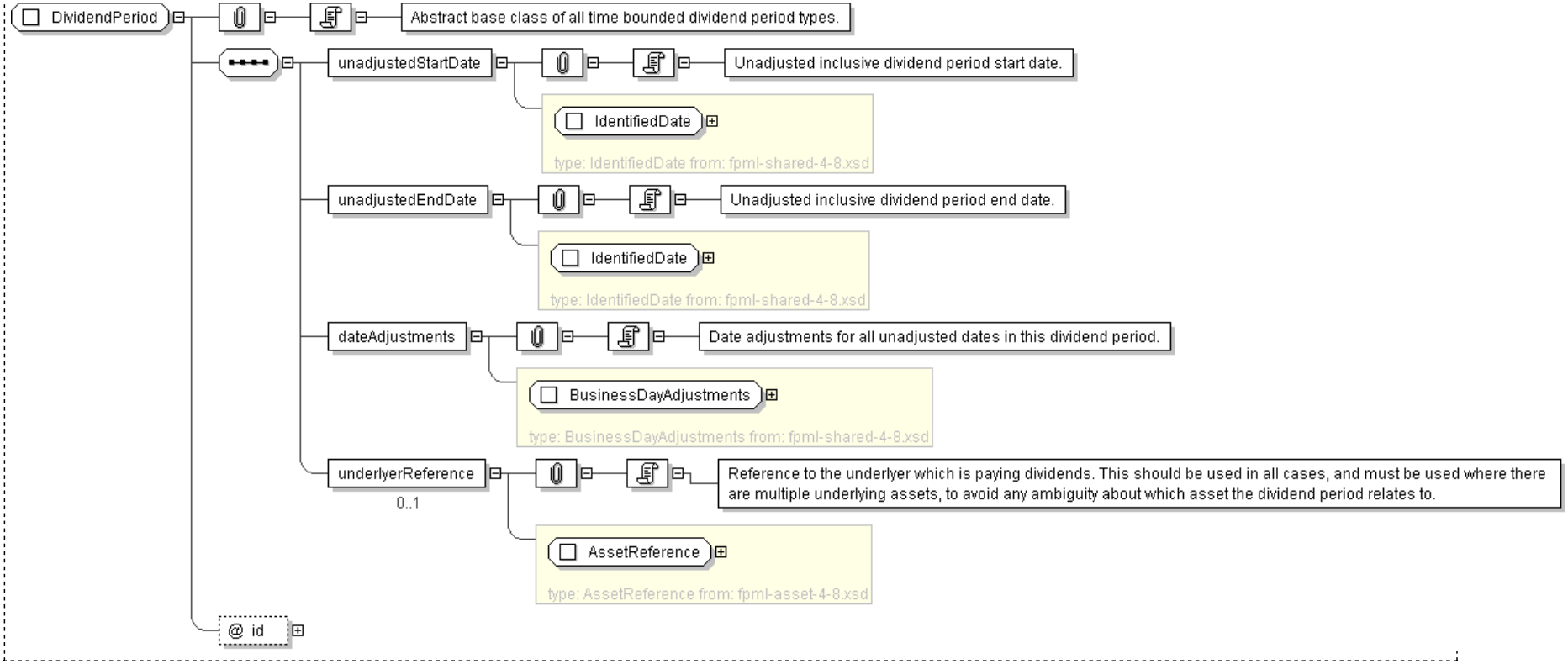
Super-types:	None
Sub-types:	• DividendPeriodDividend (by extension)

Name	DividendPeriod
Abstract	yes
Documentation	Abstract base class of all time bounded dividend period types.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedStartDate> IdentifiedDate </unadjustedStartDate> [1]
   'Unadjusted inclusive dividend period start date.'

   <unadjustedEndDate> IdentifiedDate </unadjustedEndDate> [1]
   'Unadjusted inclusive dividend period end date.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
   'Date adjustments for all unadjusted dates in this dividend period.'

   <underlyerReference> AssetReference </underlyerReference> [0..1]
   'Reference to the underlyer which is paying dividends. This should be used in all cases, and must be used where there are multiple
   underlying assets, to avoid any ambiguity about which asset the dividend period relates to.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPeriod" abstract="true">
  <xsd:sequence>
    <xsd:element name="unadjustedStartDate" type=" IdentifiedDate "/>
    <xsd:element name="unadjustedEndDate" type=" IdentifiedDate "/>
    <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments "/>
    <xsd:element name="underlyerReference" type=" AssetReference " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DividendPeriodDividend

[Table of contents]

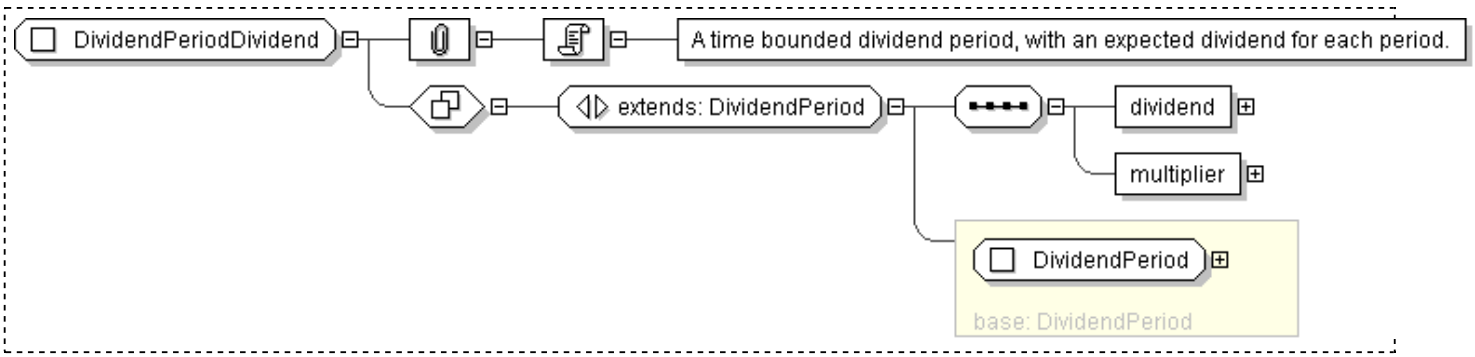
Super-types:	DividendPeriod < DividendPeriodDividend (by extension)
Sub-types:	None

Name	DividendPeriodDividend
Used by (from the same schema document)	Complex Type DividendAdjustment
Abstract	no
Documentation	A time bounded dividend period, with an expected dividend for each period.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedStartDate> IdentifiedDate </unadjustedStartDate> [1]
    'Unadjusted inclusive dividend period start date.'

   <unadjustedEndDate> IdentifiedDate </unadjustedEndDate> [1]
    'Unadjusted inclusive dividend period end date.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
    'Date adjustments for all unadjusted dates in this dividend period.'

   <underlyerReference> AssetReference </underlyerReference> [0..1]
    'Reference to the underlyer which is paying dividends. This should be used in all
    cases, and must be used where there are multiple underlying assets, to avoid any
    ambiguity about which asset the dividend period relates to.'

   <dividend> Money </dividend> [1]
    'Expected dividend in this period.'

   <multiplier> PositiveDecimal </multiplier> [1]
    'Multiplier is a percentage value which is used to produce Deviation by multiplying
    the difference between Expected Dividend and Actual Dividend Deviation = Multiplier *
    (Expected Dividend â€" Actual Dividend).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPeriodDividend">
  <xsd:complexContent>
    <xsd:extension base=" DividendPeriod ">
      <xsd:sequence>
        <xsd:element name="dividend" type=" Money "/>
        <xsd:element name="multiplier" type=" PositiveDecimal "/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityCorporateEvents

[Table of contents]

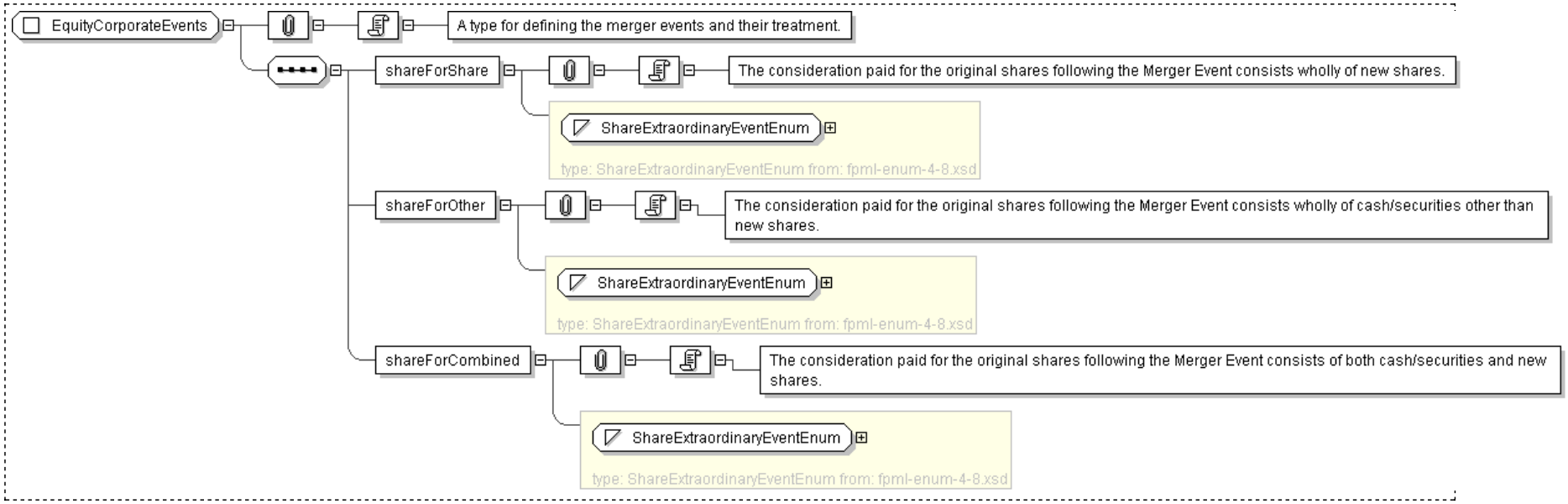
Super-types:	None
Sub-types:	None

Name	EquityCorporateEvents
Used by (from the same schema document)	Complex Type ExtraordinaryEvents , Complex Type ExtraordinaryEvents
Abstract	no
Documentation	A type for defining the merger events and their treatment.

XML Instance Representation

```
<...>
  <shareForShare> ShareExtraordinaryEventEnum </shareForShare> [1]
  'The consideration paid for the original shares following the Merger Event consists wholly of new shares.'

  <shareForOther> ShareExtraordinaryEventEnum </shareForOther> [1]
  'The consideration paid for the original shares following the Merger Event consists wholly of cash/securities other than new shares.'

  <shareForCombined> ShareExtraordinaryEventEnum </shareForCombined> [1]
  'The consideration paid for the original shares following the Merger Event consists of both cash/securities and new shares.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityCorporateEvents">
  <xsd:sequence>
    <xsd:element name="shareForShare" type=" ShareExtraordinaryEventEnum "/>
    <xsd:element name="shareForOther" type=" ShareExtraordinaryEventEnum "/>
    <xsd:element name="shareForCombined" type=" ShareExtraordinaryEventEnum "/>
```

```
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityPremium

[Table of contents]

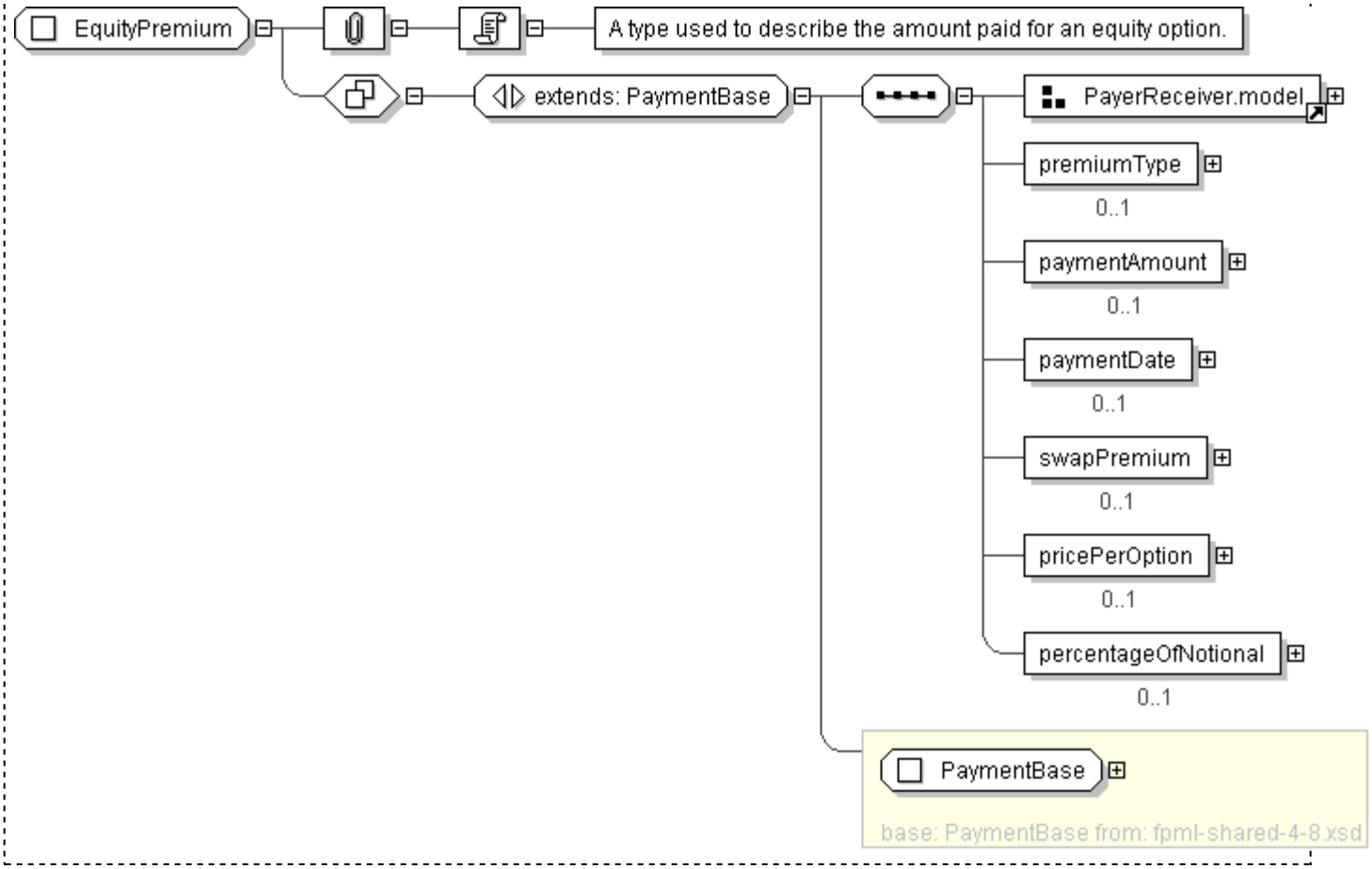
Super-types:	PaymentBase < EquityPremium (by extension)
Sub-types:	None

Name	EquityPremium
Abstract	no
Documentation	A type used to describe the amount paid for an equity option.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <premiumType> PremiumTypeEnum </premiumType> [0..1]
   'Forward start Premium type'

   <paymentAmount> Money </paymentAmount> [0..1]
   'The currency amount of the payment.'

   <paymentDate> AdjustableDate </paymentDate> [0..1]
   'The payment date. This date is subject to adjustment in accordance with any
   applicable business day convention.'

   <swapPremium> xsd:boolean </swapPremium> [0..1]
   'Specifies whether or not the premium is to be paid in the style of payments
   under an interest rate swap contract.'

   <pricePerOption> Money </pricePerOption> [0..1]
   'The amount of premium to be paid expressed as a function of the number of
   options.'

   <percentageOfNotional> xsd:decimal </percentageOfNotional> [0..1]
   'The amount of premium to be paid expressed as a percentage of the notional
   value of the transaction. A percentage of 5% would be expressed as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityPremium">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="premiumType" type=" PremiumTypeEnum " minOccurs="0"/>
        <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
        <xsd:element name="paymentDate" type=" AdjustableDate " minOccurs="0"/>
        <xsd:element name="swapPremium" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="pricePerOption" type=" Money " minOccurs="0"/>
        <xsd:element name="percentageOfNotional" type=" xsd:decimal "
        minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityStrike

[Table of contents]

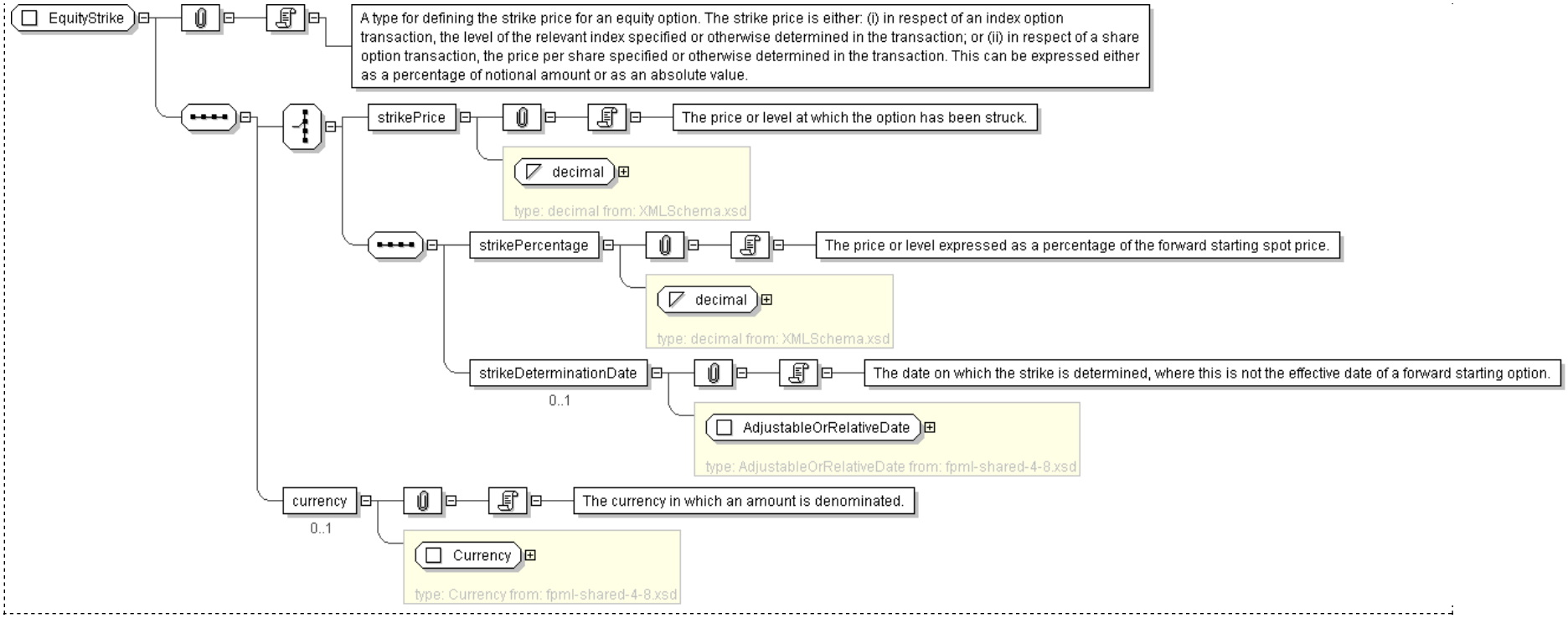
Super-types:	None
Sub-types:	None

Name	EquityStrike
Abstract	no
Documentation	A type for defining the strike price for an equity option. The strike price is either: (i) in respect of an index option transaction, the level of the relevant index specified or otherwise determined in the transaction; or (ii) in respect of a share option transaction, the price per share specified or otherwise determined in the transaction. This can be expressed either as a percentage of notional amount or as an absolute value.

XML Instance Representation

```
<...>
  Start Choice [1]
    <strikePrice> xsd:decimal </strikePrice> [1]
    'The price or level at which the option has been struck.'

    <strikePercentage> xsd:decimal </strikePercentage> [1]
    'The price or level expressed as a percentage of the forward starting spot price.'

    <strikeDeterminationDate> AdjustableOrRelativeDate </strikeDeterminationDate> [0..1]
    'The date on which the strike is determined, where this is not the effective date of a forward starting option.'

  End Choice
  <currency> Currency </currency> [0..1]
  'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityStrike">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="strikePrice" type=" xsd:decimal "/>
      <xsd:sequence>
        <xsd:element name="strikePercentage" type=" xsd:decimal "/>
        <xsd:element name="strikeDeterminationDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      </xsd:sequence>
    </xsd:choice>
    <xsd:element name="currency" type=" Currency " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquityValuation

[Table of contents]

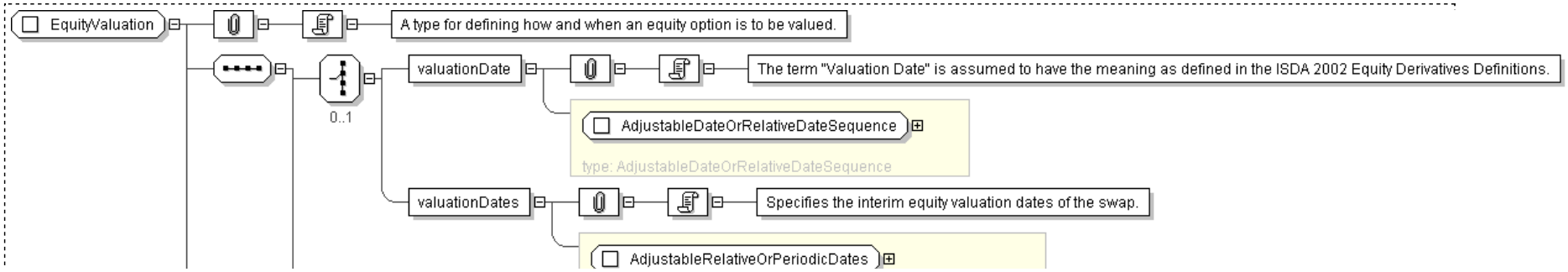
Super-types:	None
Sub-types:	None

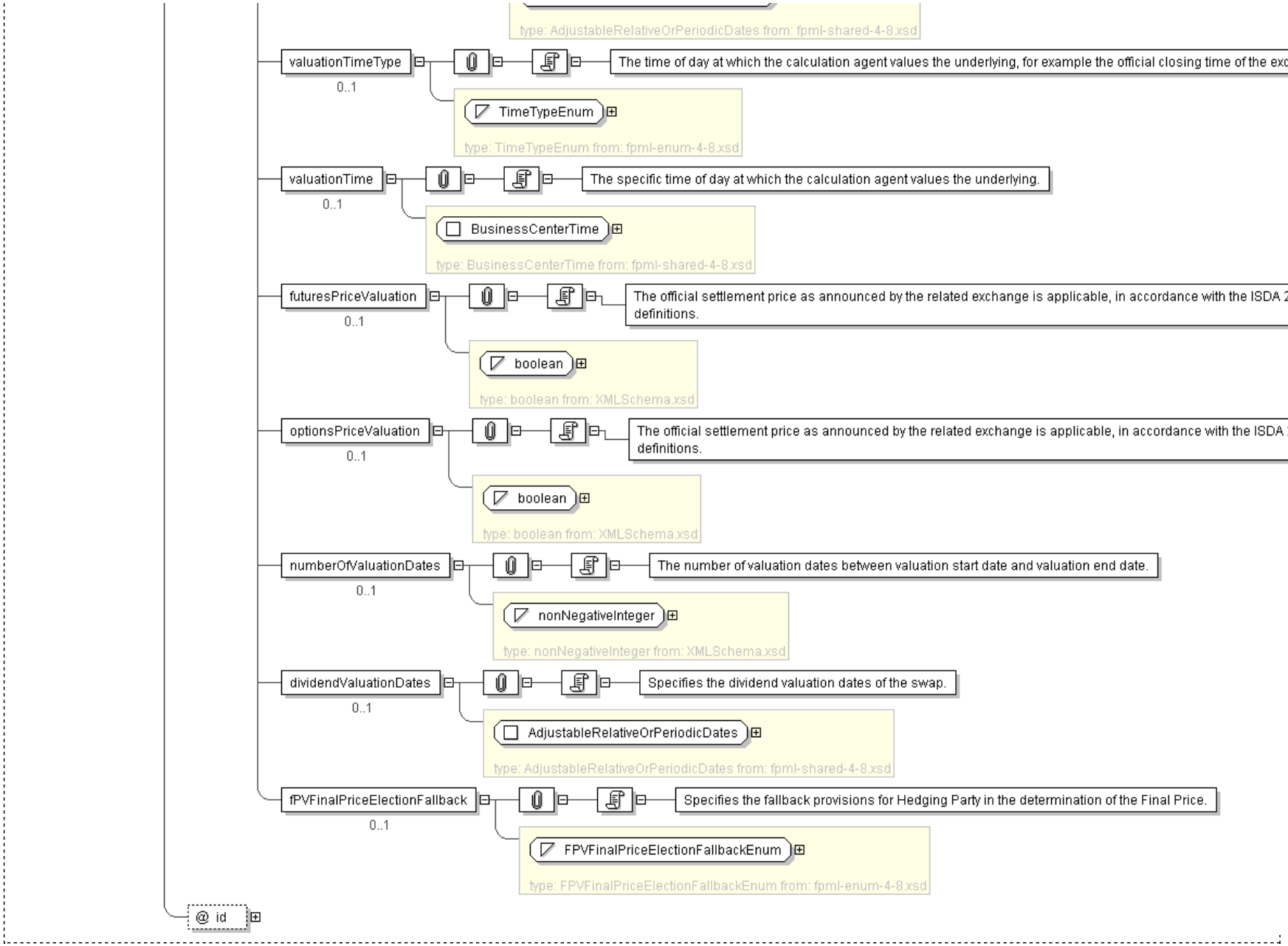
Name	EquityValuation
Used by (from the same schema document)	Complex Type DeprecatedVarianceLeg , Complex Type DirectionalLegUnderlyerValuation , Complex Type ReturnLegValuationPrice
Abstract	no
Documentation	A type for defining how and when an equity option is to be valued.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [0..1]
      <valuationDate> AdjustableDateOrRelativeDateSequence </valuationDate> [1]
      'The term \"Valuation Date\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

      <valuationDates> AdjustableRelativeOrPeriodicDates </valuationDates> [1]
      'Specifies the interim equity valuation dates of the swap.'

   End Choice
   <valuationTimeType> TimeTypeEnum </valuationTimeType> [0..1]
   'The time of day at which the calculation agent values the underlying, for example the official closing time of the exchange.'

   <valuationTime> BusinessCenterTime </valuationTime> [0..1]
   'The specific time of day at which the calculation agent values the underlying.'

   <futuresPriceValuation> xsd:boolean </futuresPriceValuation> [0..1]
   'The official settlement price as announced by the related exchange is applicable, in accordance with the ISDA 2002 definitions.'

   <optionsPriceValuation> xsd:boolean </optionsPriceValuation> [0..1]
   'The official settlement price as announced by the related exchange is applicable, in accordance with the ISDA 2002 definitions.'

   <numberOfValuationDates> xsd:nonNegativeInteger </numberOfValuationDates> [0..1]
   'The number of valuation dates between valuation start date and valuation end date.'

   <dividendValuationDates> AdjustableRelativeOrPeriodicDates </dividendValuationDates> [0..1]
   'Specifies the dividend valuation dates of the swap.'

   <fPVFinalPriceElectionFallback> FPVFinalPriceElectionFallbackEnum </fPVFinalPriceElectionFallback> [0..1]
   'Specifies the fallback provisions for Hedging Party in the determination of the Final Price.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="EquityValuation">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="valuationDate" type=" AdjustableDateOrRelativeDateSequence "/>
      <xsd:element name="valuationDates" type=" AdjustableRelativeOrPeriodicDates "/>
    </xsd:choice>
    <xsd:element name="valuationTimeType" type=" TimeTypeEnum " minOccurs="0"/>
    <xsd:element name="valuationTime" type=" BusinessCenterTime " minOccurs="0"/>
    <xsd:element name="futuresPriceValuation" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="optionsPriceValuation" type=" xsd:boolean " minOccurs="0"/>
```

```
      <xsd:element name="numberOfValuationDates" type=" xsd:nonNegativeInteger " minOccurs="0"/>
      <xsd:element name="dividendValuationDates" type=" AdjustableRelativeOrPeriodicDates " minOccurs="0"/>
      <xsd:element name="fPVFinalPriceElectionFallback" type=" FPVFinalPriceElectionFallbackEnum " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExtraordinaryEvents

[Table of contents]

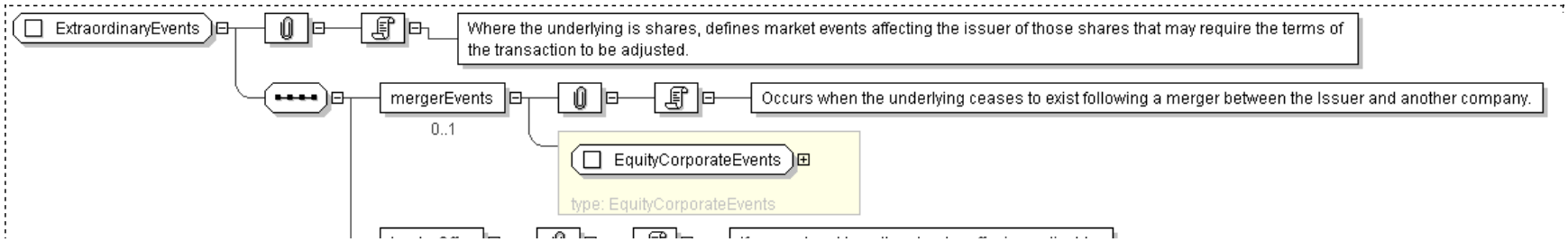
Super-types:	None
Sub-types:	None

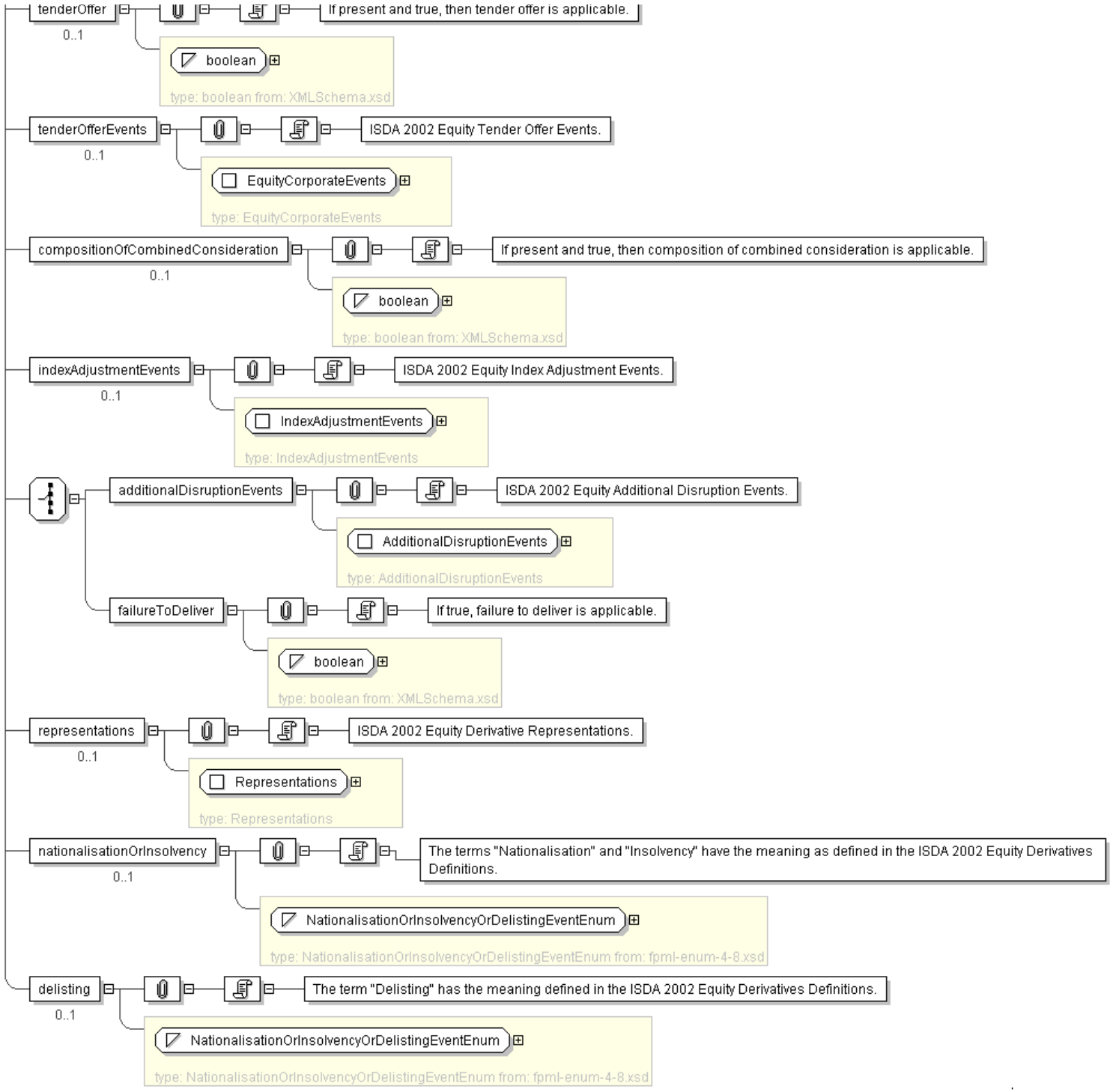
Name	ExtraordinaryEvents
Used by (from the same schema document)	Complex Type NettedSwapBase , Complex Type ReturnSwap
Abstract	no
Documentation	Where the underlying is shares, defines market events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.

XML Instance Representation

```
<...>
  <mergerEvents> EquityCorporateEvents </mergerEvents> [0..1]
  'Occurs when the underlying ceases to exist following a merger between the Issuer and another company.'

  <tenderOffer> xsd:boolean </tenderOffer> [0..1]
  'If present and true, then tender offer is applicable.'

  <tenderOfferEvents> EquityCorporateEvents </tenderOfferEvents> [0..1]
  'ISDA 2002 Equity Tender Offer Events.'

  <compositionOfCombinedConsideration> xsd:boolean </compositionOfCombinedConsideration> [0..1]
  'If present and true, then composition of combined consideration is applicable.'

  <indexAdjustmentEvents> IndexAdjustmentEvents </indexAdjustmentEvents> [0..1]
  'ISDA 2002 Equity Index Adjustment Events.'

  Start Choice [1]
    <additionalDisruptionEvents> AdditionalDisruptionEvents </additionalDisruptionEvents> [1]
    'ISDA 2002 Equity Additional Disruption Events.'

    <failureToDeliver> xsd:boolean </failureToDeliver> [1]
    'If true, failure to deliver is applicable.'

  End Choice
  <representations> Representations </representations> [0..1]
  'ISDA 2002 Equity Derivative Representations.'

  <nationalisationOrInsolvency> NationalisationOrInsolvencyOrDelistingEventEnum </nationalisationOrInsolvency> [0..1]
  'The terms \"Nationalisation\" and \"Insolvency\" have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.'

  <delisting> NationalisationOrInsolvencyOrDelistingEventEnum </delisting> [0..1]
  'The term \"Delisting\" has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="ExtraordinaryEvents">
```

```
  <xsd:sequence>
    <xsd:element name="mergerEvents" type=" EquityCorporateEvents " minOccurs="0"/>
    <xsd:element name="tenderOffer" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="tenderOfferEvents" type=" EquityCorporateEvents " minOccurs="0"/>
    <xsd:element name="compositionOfCombinedConsideration" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="indexAdjustmentEvents" type=" IndexAdjustmentEvents " minOccurs="0"/>
    <xsd:choice>
      <xsd:element name="additionalDisruptionEvents" type=" AdditionalDisruptionEvents "/>
      <xsd:element name="failureToDeliver" type=" xsd:boolean "/>
    </xsd:choice>
    <xsd:element name="representations" type=" Representations " minOccurs="0"/>
    <xsd:element name="nationalisationOrInsolvency" type=" NationalisationOrInsolvencyOrDelistingEventEnum " minOccurs="0"/>
    <xsd:element name="delisting" type=" NationalisationOrInsolvencyOrDelistingEventEnum " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingRateCalculationReference

[Table of contents]

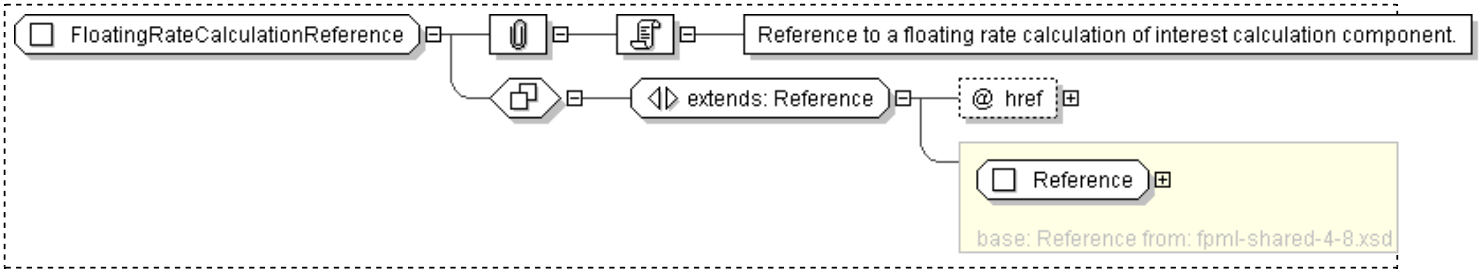
Super-types:	Reference < FloatingRateCalculationReference (by extension)
Sub-types:	None

Name	FloatingRateCalculationReference
Used by (from the same schema document)	Complex Type CompoundingRate
Abstract	no
Documentation	Reference to a floating rate calculation of interest calculation component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingRateCalculationReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="floatingRateCalculation"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IndexAdjustmentEvents

[Table of contents]

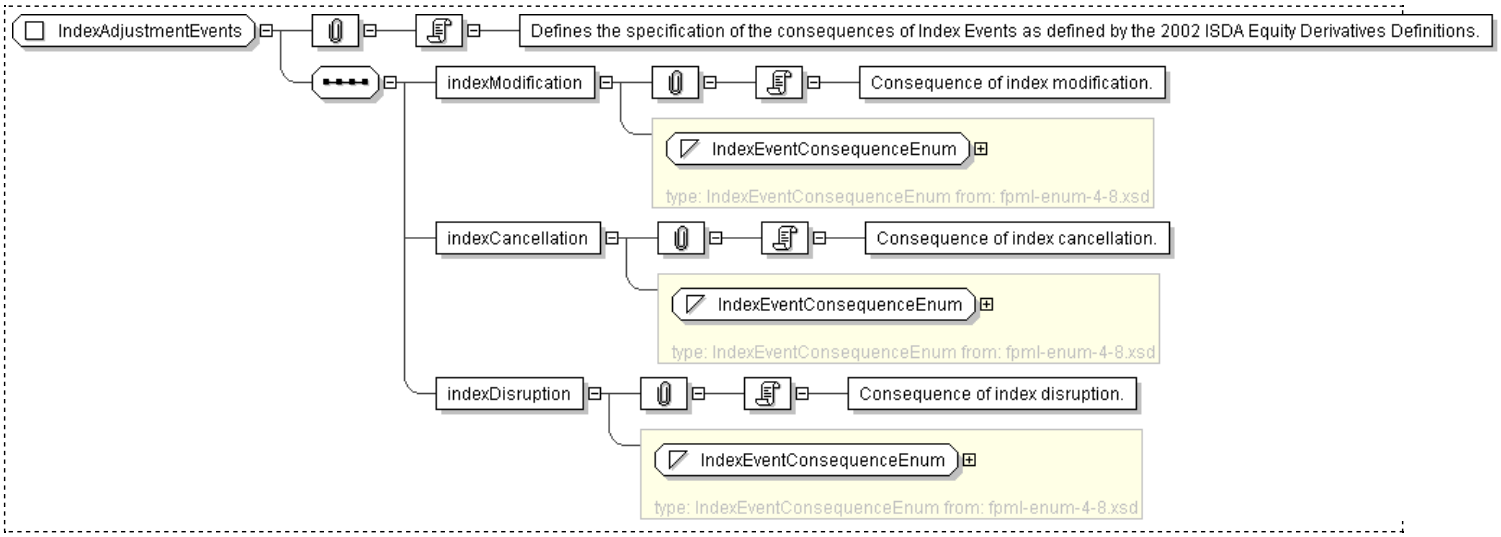
Super-types:	None
Sub-types:	None

Name	IndexAdjustmentEvents
Used by (from the same schema document)	Complex Type ExtraordinaryEvents
Abstract	no
Documentation	Defines the specification of the consequences of Index Events as defined by the 2002 ISDA Equity Derivatives Definitions.

XML Instance Representation

```
<...>
   <indexModification> IndexEventConsequenceEnum </indexModification> [1]
   'Consequence of index modification.'

   <indexCancellation> IndexEventConsequenceEnum </indexCancellation> [1]
   'Consequence of index cancellation.'

   <indexDisruption> IndexEventConsequenceEnum </indexDisruption> [1]
   'Consequence of index disruption.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexAdjustmentEvents">
   <xsd:sequence>
      <xsd:element name="indexModification" type=" IndexEventConsequenceEnum "/>
      <xsd:element name="indexCancellation" type=" IndexEventConsequenceEnum "/>
      <xsd:element name="indexDisruption" type=" IndexEventConsequenceEnum "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestCalculation

[Table of contents]

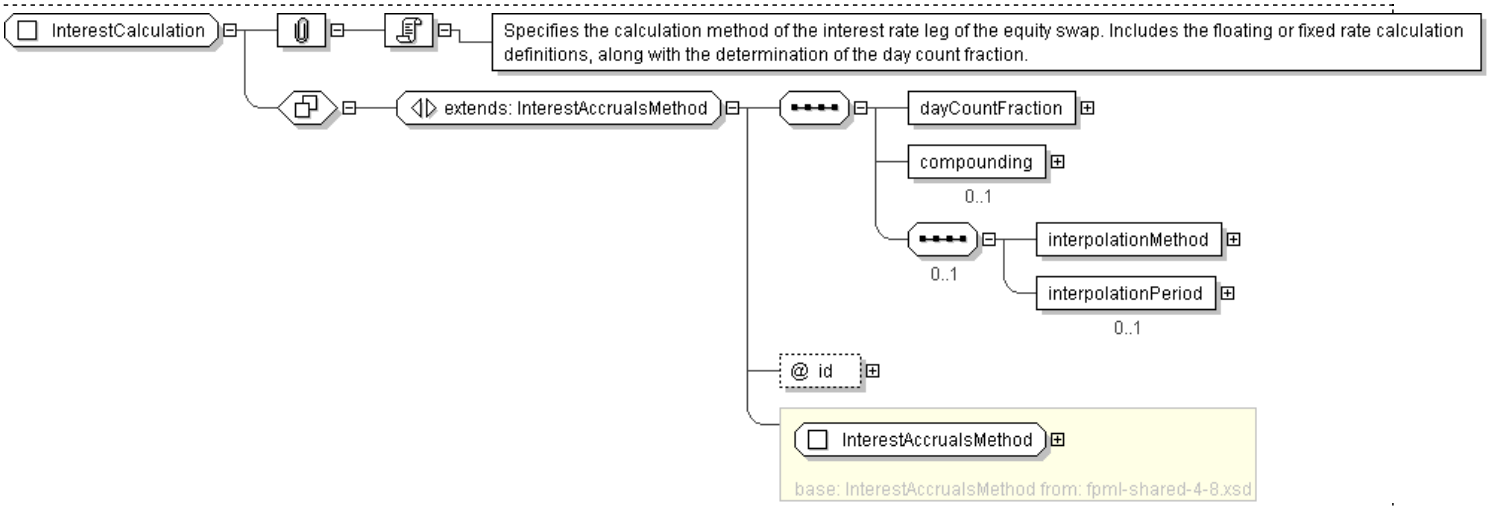
Super-types:	InterestAccrualsMethod < InterestCalculation (by extension)
Sub-types:	None

Name	InterestCalculation
Used by (from the same schema document)	Complex Type InterestLeg
Abstract	no
Documentation	Specifies the calculation method of the interest rate leg of the equity swap. Includes the floating or fixed rate calculation definitions, along with the determination of the day count fraction.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <floatingRateCalculation> FloatingRateCalculation </floatingRateCalculation> [1]
      'The floating rate calculation definitions'

      <fixedRate> xsd:decimal </fixedRate> [1]
      'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be
      represented as 0.05.'

   End Choice
   <dayCountFraction> DayCountFraction </dayCountFraction> [1]
   'The day count fraction.'

   <compounding> Compounding </compounding> [0..1]
   'Defines compounding rates on the Interest Leg.'

   Start Sequence [0..1]
      <interpolationMethod> InterpolationMethod </interpolationMethod> [1]
      'Specifies the type of interpolation used.'

      <interpolationPeriod> InterpolationPeriodEnum </interpolationPeriod> [0..1]
      'Defines applicable periods for interpolation.'

   End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestCalculation">
  <xsd:complexContent>
    <xsd:extension base=" InterestAccrualsMethod ">
      <xsd:sequence>
        <xsd:element name="dayCountFraction" type=" DayCountFraction "/>
        <xsd:element name="compounding" type=" Compounding " minOccurs="0"/>
        <xsd:sequence minOccurs="0">
          <xsd:element name="interpolationMethod" type=" InterpolationMethod "/>
          <xsd:element name="interpolationPeriod" type=" InterpolationPeriodEnum " minOccurs="0"/>
        </xsd:sequence>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID "/>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestLeg

[Table of contents]

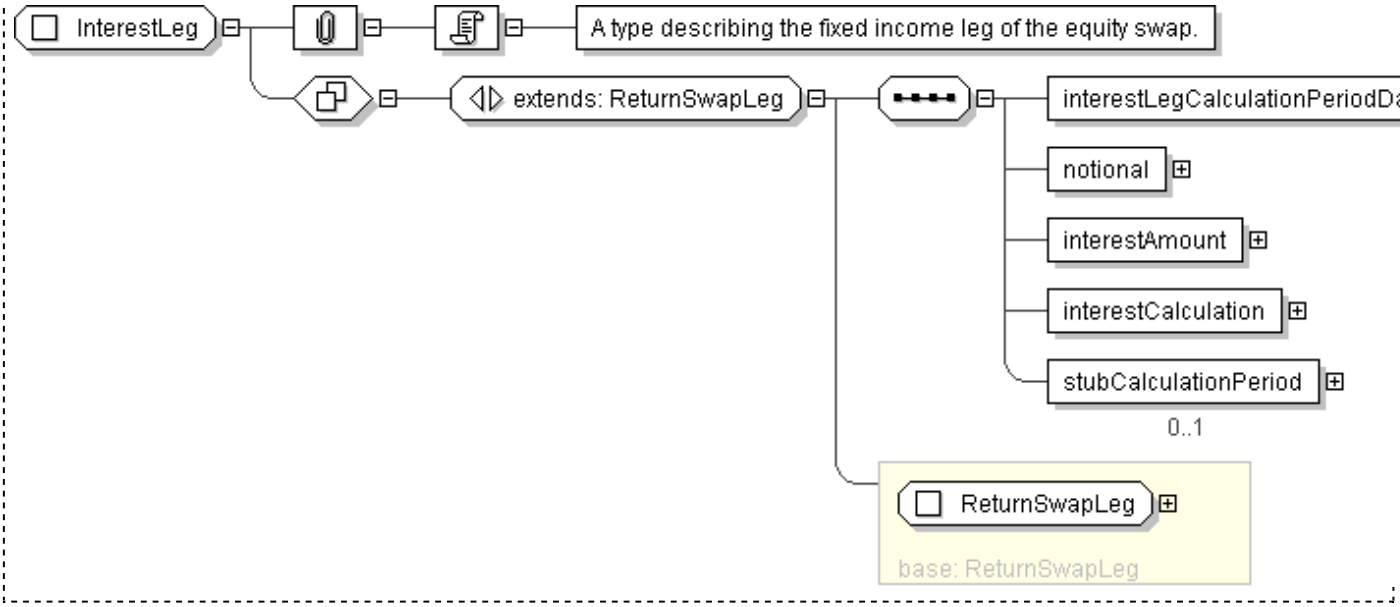
Super-types:	Leg < ReturnSwapLeg (by extension) < InterestLeg (by extension)
Sub-types:	None

Name	InterestLeg
Used by (from the same schema document)	Element interestLeg
Abstract	no
Documentation	A type describing the fixed income leg of the equity swap.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version.
   Frequency at which this leg pays.'

   <interestLegCalculationPeriodDates> InterestLegCalculationPeriodDates
   </interestLegCalculationPeriodDates> [1]
   'Component that holds the various dates used to specify the interest leg of the
   equity swap. It is used to define the InterestPeriodDates identifyer.'

   <notional> ReturnSwapNotional </notional> [1]
   'Specifies the notional of a return type swap. When used in the equity leg, the
   definition will typically combine the actual amount (using the notional component
   defined by the FpML industry group) and the determination method. When used in
   the interest leg, the definition will typically point to the definition of the
   equity leg.'

   <interestAmount> LegAmount </interestAmount> [1]
   'Specifies, in relation to each Interest Payment Date, the amount to which the
   Interest Payment Date relates. Unless otherwise specified, this term has the
   meaning defined in the ISDA 2000 ISDA Definitions.'

   <interestCalculation> InterestCalculation </interestCalculation> [1]
   'Specifies the calculation method of the interest rate leg of the equity swap.
   Includes the floating or fixed rate calculation definitions, along with the
   determination of the day count fraction.'

   <stubCalculationPeriod> StubCalculationPeriod </stubCalculationPeriod> [0..1]
   'Specifies the stub calculation period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLeg">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="interestLegCalculationPeriodDates" type="
            InterestLegCalculationPeriodDates "/>
            <xsd:element name="notional" type=" ReturnSwapNotional "/>
            <xsd:element name="interestAmount" type=" LegAmount "/>
            <xsd:element name="interestCalculation" type=" InterestCalculation "/>
            <xsd:element name="stubCalculationPeriod" type=" StubCalculationPeriod "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestLegCalculationPeriodDates

[Table of contents]

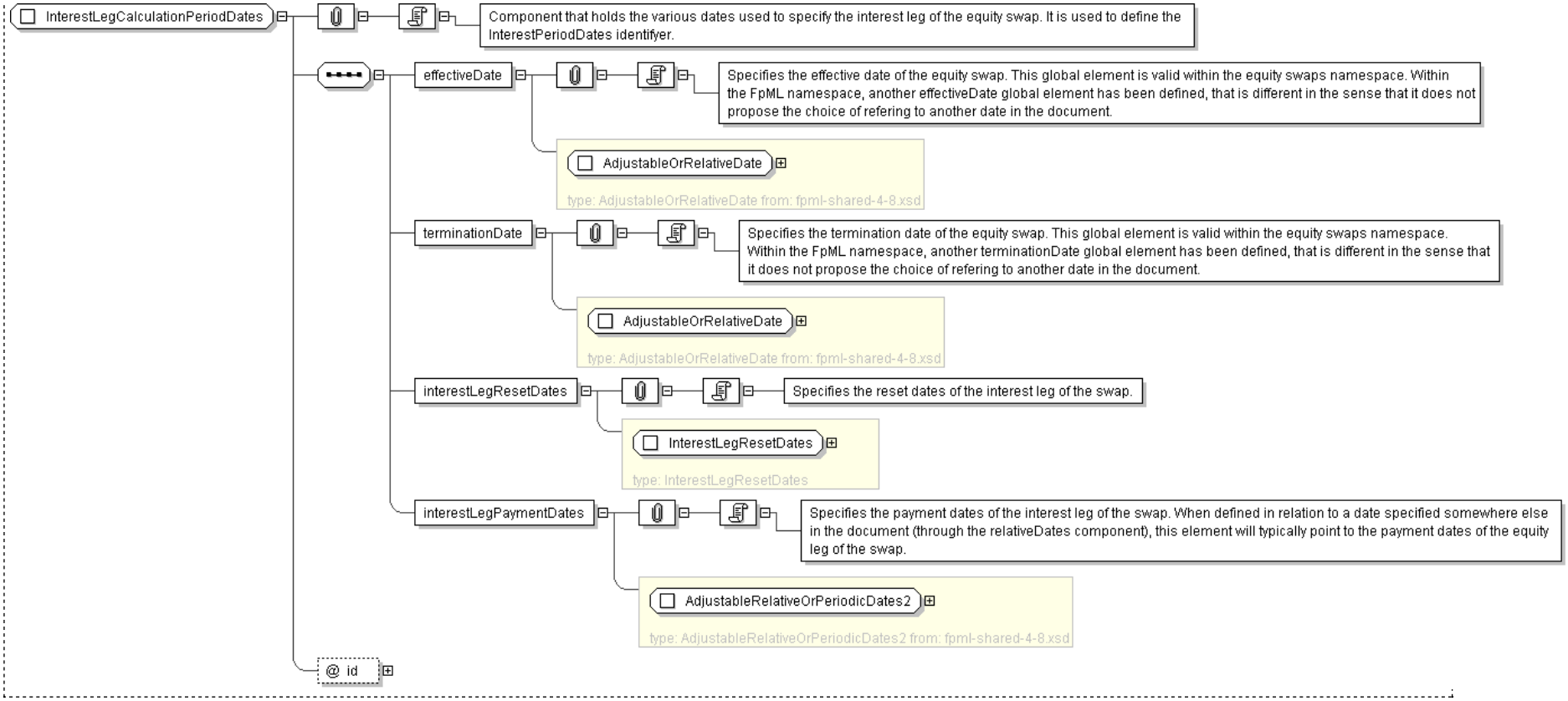
Super-types:	None
Sub-types:	None

Name	InterestLegCalculationPeriodDates
Used by (from the same schema document)	Complex Type InterestLeg
Abstract	no
Documentation	Component that holds the various dates used to specify the interest leg of the equity swap. It is used to define the InterestPeriodDates identifyer.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
    'Specifies the effective date of the equity swap. This global element is valid within the equity swaps namespace. Within the FpML namespace, another
    effectiveDate global element has been defined, that is different in the sense that it does not propose the choice of refering to another date in the
    document.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
    'Specifies the termination date of the equity swap. This global element is valid within the equity swaps namespace. Within the FpML namespace, another
    terminationDate global element has been defined, that is different in the sense that it does not propose the choice of refering to another date in the
    document.'

   <interestLegResetDates> InterestLegResetDates </interestLegResetDates> [1]
    'Specifies the reset dates of the interest leg of the swap.'

   <interestLegPaymentDates> AdjustableRelativeOrPeriodicDates2 </interestLegPaymentDates> [1]
    'Specifies the payment dates of the interest leg of the swap. When defined in relation to a date specified somewhere else in the document (through the
    relativeDates component), this element will typically point to the payment dates of the equity leg of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLegCalculationPeriodDates">
  <xsd:sequence>
    <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
    <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
    <xsd:element name="interestLegResetDates" type=" InterestLegResetDates "/>
    <xsd:element name="interestLegPaymentDates" type=" AdjustableRelativeOrPeriodicDates2 "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestLegCalculationPeriodDatesReference

[Table of contents]

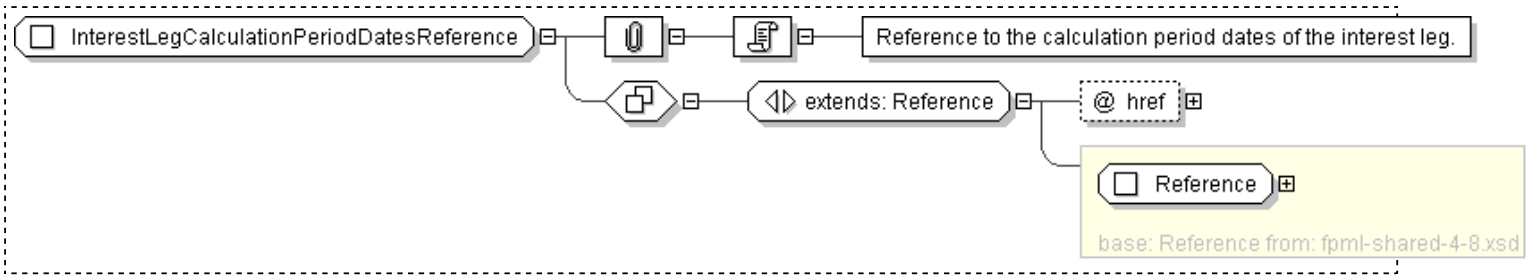
Super-types:	Reference < InterestLegCalculationPeriodDatesReference (by extension)
Sub-types:	None

Name	InterestLegCalculationPeriodDatesReference
Used by (from the same schema document)	Complex Type InterestLegResetDates
Abstract	no
Documentation	Reference to the calculation period dates of the interest leg.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLegCalculationPeriodDatesReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="InterestLegCalculationPeriodDates"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestLegResetDates

[Table of contents]

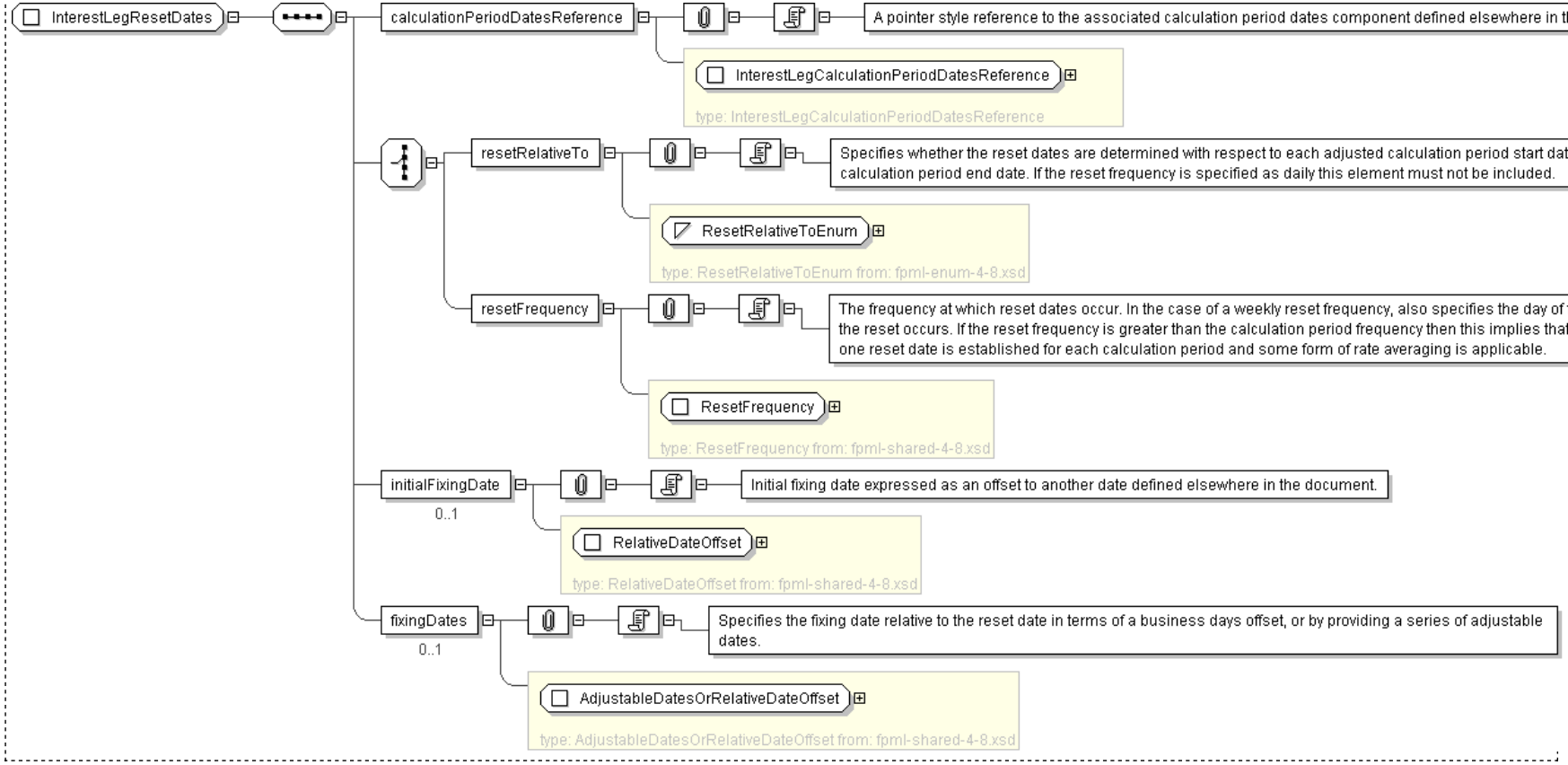
Super-types:	None
Sub-types:	None

Name	InterestLegResetDates
Used by (from the same schema document)	Complex Type InterestLegCalculationPeriodDates
Abstract	no

XML Instance Representation

```
<...>
  <calculationPeriodDatesReference> InterestLegCalculationPeriodDatesReference </calculationPeriodDatesReference> [1]
  'A pointer style reference to the associated calculation period dates component defined elsewhere in the document.'

  Start Choice [1]
    <resetRelativeTo> ResetRelativeToEnum </resetRelativeTo> [1]
    'Specifies whether the reset dates are determined with respect to each adjusted calculation period start date or adjusted calculation period end
    date. If the reset frequency is specified as daily this element must not be included.'

    <resetFrequency> ResetFrequency </resetFrequency> [1]
    'The frequency at which reset dates occur. In the case of a weekly reset frequency, also specifies the day of the week that the reset occurs. If
    the reset frequency is greater than the calculation period frequency then this implies that more than one reset date is established for each
    calculation period and some form of rate averaging is applicable.'

  End Choice
  <initialFixingDate> RelativeDateOffset </initialFixingDate> [0..1]
  'Initial fixing date expressed as an offset to another date defined elsewhere in the document.'

  <fixingDates> AdjustableDatesOrRelativeDateOffset </fixingDates> [0..1]
  'Specifies the fixing date relative to the reset date in terms of a business days offset, or by providing a series of adjustable dates.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLegResetDates">
  <xsd:sequence>
    <xsd:element name="calculationPeriodDatesReference" type=" InterestLegCalculationPeriodDatesReference "/>
    <xsd:choice>
      <xsd:element name="resetRelativeTo" type=" ResetRelativeToEnum "/>
      <xsd:element name="resetFrequency" type=" ResetFrequency "/>
    </xsd:choice>
    <xsd:element name="initialFixingDate" type=" RelativeDateOffset " minOccurs="0"/>
    <xsd:element name="fixingDates" type=" AdjustableDatesOrRelativeDateOffset " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LegAmount

[Table of contents]

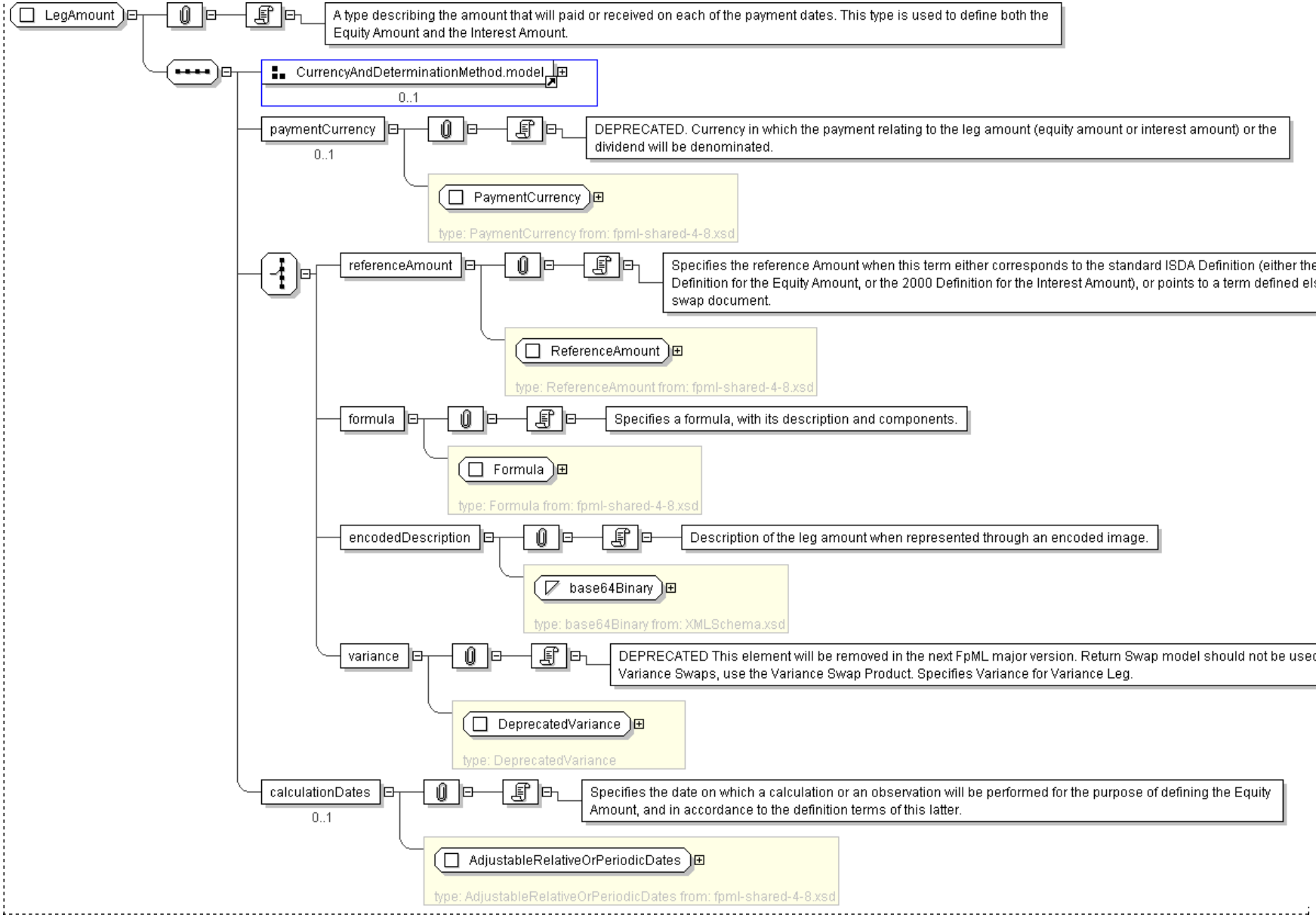
Super-types:	None
Sub-types:	• ReturnSwapAmount (by extension) ◦ DeprecatedVarianceAmount (by extension)

Name	LegAmount
Used by (from the same schema document)	Complex Type InterestLeg
Abstract	no
Documentation	A type describing the amount that will paid or received on each of the payment dates. This type is used to define both the Equity Amount and the Interest Amount.

XML Instance Representation

```
<...>
  Start Group: CurrencyAndDeterminationMethod.model [0..1]
    Start Choice [1]
      <currency> Currency </currency> [1]
      'The currency in which an amount is denominated.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <currencyReference> IdentifiedCurrencyReference </currencyReference> [1]
      'Reference to a currency defined elsewhere in the document'

    End Choice
  End Group: CurrencyAndDeterminationMethod.model
  <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
  'DEPRECATED. Currency in which the payment relating to the leg amount (equity amount or interest amount) or the dividend will be denominated.'

  Start Choice [1]
    <referenceAmount> ReferenceAmount </referenceAmount> [1]
    'Specifies the reference Amount when this term either corresponds to the standard ISDA Definition (either the 2002 Equity Definition for
    the Equity Amount, or the 2000 Definition for the Interest Amount), or points to a term defined elsewhere in the swap document.'

    <formula> Formula </formula> [1]
    'Specifies a formula, with its description and components.'

    <encodedDescription> xsd:base64Binary </encodedDescription> [1]
    'Description of the leg amount when represented through an encoded image.'

    <variance> DeprecatedVariance </variance> [1]
    'DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the
    Variance Swap Product. Specifies Variance for Variance Leg.'

  End Choice
  <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
  'Specifies the date on which a calculation or an observation will be performed for the purpose of defining the Equity Amount, and in
  accordance to the definition terms of this latter.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LegAmount">
  <xsd:sequence>
    <xsd:group ref=" CurrencyAndDeterminationMethod.model " minOccurs="0"/>
    <xsd:element name="paymentCurrency" type=" PaymentCurrency " minOccurs="0" deprecated="true" deprecatedReason="The model is wrong since it has
    an intradocument reference that is not clear. Current PaymentCurrency model and elements using this type are deprecated. Instead, the choice
    above between currency, determinationMethod, and currencyReference (of type CurrencyReference) should be used."/>
    <xsd:choice>
      <xsd:element name="referenceAmount" type=" ReferenceAmount "/>
      <xsd:element name="formula" type=" Formula "/>
      <xsd:element name="encodedDescription" type=" xsd:base64Binary "/>
      <xsd:element name="variance" type=" DeprecatedVariance " deprecated="true" deprecatedReason="Return Swap model should not be used for
      Variance Swaps, use the Variance Swap Product"/>
    </xsd:choice>
    <xsd:element name="calculationDates" type=" AdjustableRelativeOrPeriodicDates " minOccurs="0"/>
```

```
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LegId

[Table of contents]

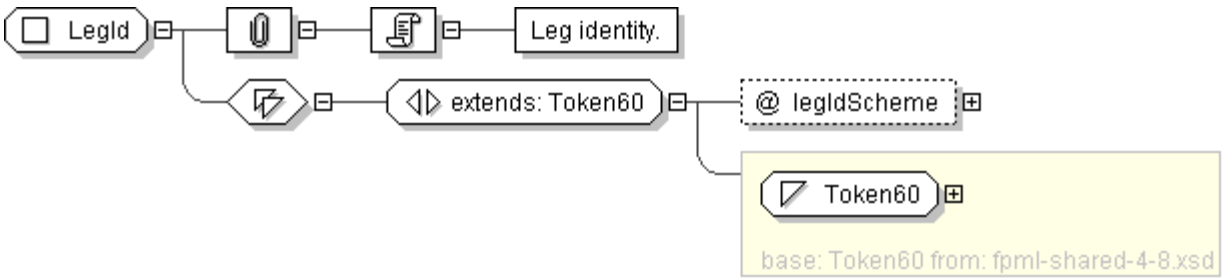
Super-types:	Token60 < LegId (by extension)
Sub-types:	None

Name	LegId
Used by (from the same schema document)	Complex Type LegIdentifier
Abstract	no
Documentation	Leg identity.

XML Instance Representation

```
<...
 legIdScheme=" xsd:anyURI [1]">
   Token60
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LegId">
   <xsd:simpleContent>
      <xsd:extension base=" Token60 ">
         <xsd:attribute name="legIdScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LegIdentifier

[Table of contents]

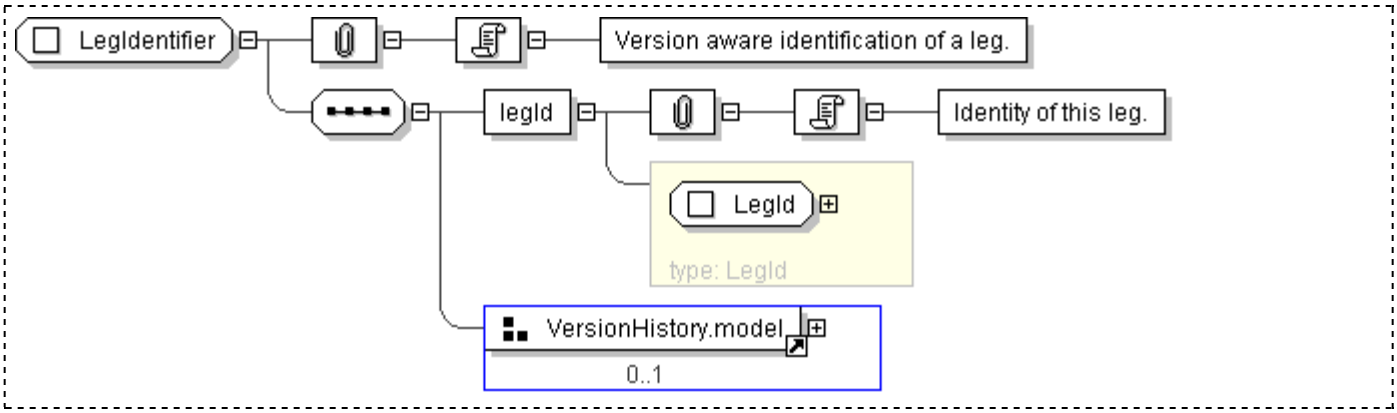
Super-types:	None
Sub-types:	None

Name	LegIdentifier
Used by (from the same schema document)	Complex Type DirectionalLeg
Abstract	no
Documentation	Version aware identification of a leg.

XML Instance Representation

```
<...>
   <legId> LegId </legId> [1]
    'Identity of this leg.'

   Start Group: VersionHistory.model [0..1]
      <version> xsd:nonNegativeInteger </version> [1]
       'The version number'

      <effectiveDate> IdentifiedDate </effectiveDate> [0..1]
       'Optionally it is possible to specify a version effective date when a
       versionId is supplied.'

   End Group: VersionHistory.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LegIdentifier">
   <xsd:sequence>
      <xsd:element name="legId" type=" LegId "/>
      <xsd:group ref=" VersionHistory.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MakeWholeProvisions

[Table of contents]

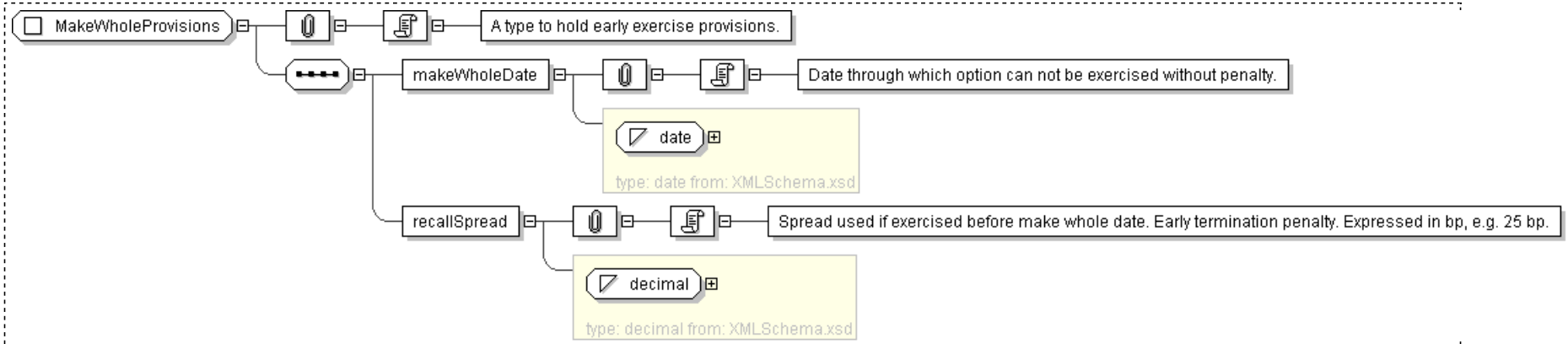
Super-types:	None
Sub-types:	None

Name	MakeWholeProvisions
Abstract	no
Documentation	A type to hold early exercise provisions.

XML Instance Representation

```
<...>
  <makeWholeDate> xsd:date </makeWholeDate> [1]
   'Date through which option can not be exercised without penalty.'

  <recallSpread> xsd:decimal </recallSpread> [1]
   'Spread used if exercised before make whole date. Early termination penalty. Expressed in bp, e.g. 25 bp.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MakeWholeProvisions">
  <xsd:sequence>
    <xsd:element name="makeWholeDate" type=" xsd:date "/>
    <xsd:element name="recallSpread" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NettedSwapBase

[Table of contents]

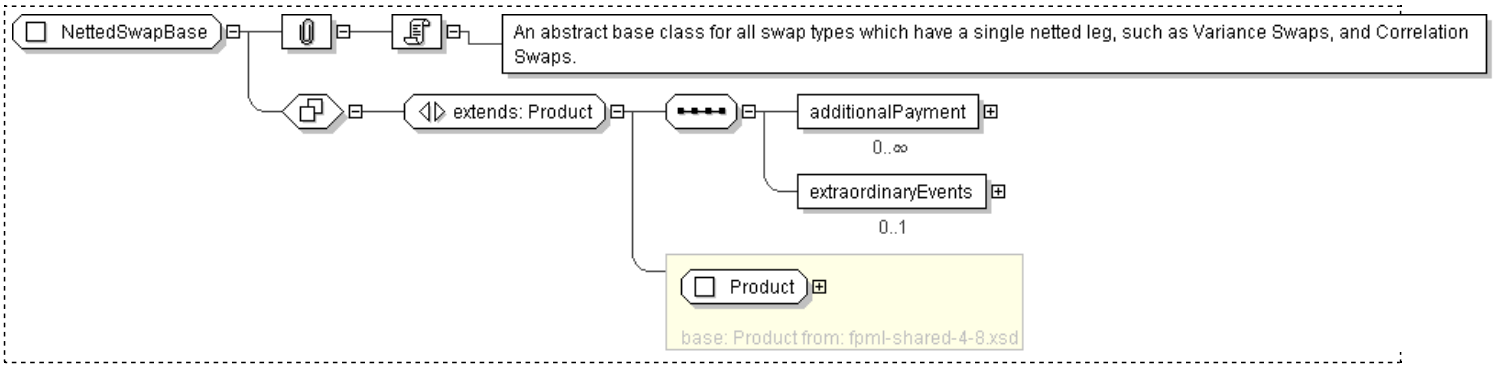
Super-types:	Product < NettedSwapBase (by extension)
Sub-types:	None

Name	NettedSwapBase
Abstract	yes
Documentation	An abstract base class for all swap types which have a single netted leg, such as Variance Swaps, and Correlation Swaps.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding
    scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated
    with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
    'Specifies additional payment(s) between the principal parties to the netted swap.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
    'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the
    terms of the transaction to be adjusted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NettedSwapBase" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="additionalPayment" type=" ClassifiedPayment " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionFeatures

[Table of contents]

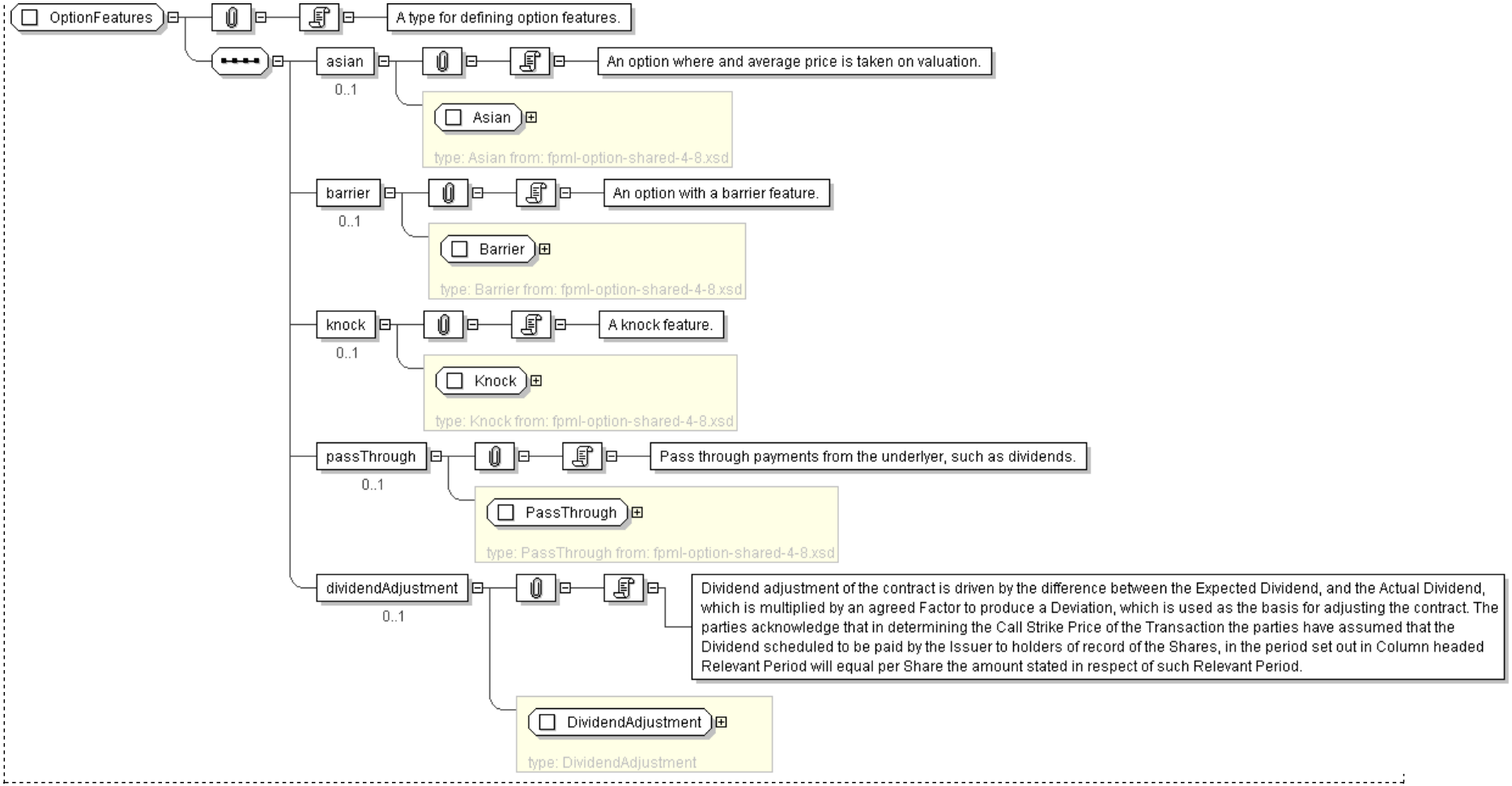
Super-types:	None
Sub-types:	None

Name	OptionFeatures
Used by (from the same schema document)	Model Group Feature.model
Abstract	no
Documentation	A type for defining option features.

XML Instance Representation

```
<...>
   <asian> Asian </asian> [0..1]
   'An option where and average price is taken on valuation.'

   <barrier> Barrier </barrier> [0..1]
   'An option with a barrier feature.'

   <knock> Knock </knock> [0..1]
   'A knock feature.'

   <passThrough> PassThrough </passThrough> [0..1]
   'Pass through payments from the underlyer, such as dividends.'

   <dividendAdjustment> DividendAdjustment </dividendAdjustment> [0..1]
   'Dividend adjustment of the contract is driven by the difference between the Expected Dividend, and the Actual Dividend, which is
   multiplied by an agreed Factor to produce a Deviation, which is used as the basis for adjusting the contract. The parties acknowledge
   that in determining the Call Strike Price of the Transaction the parties have assumed that the Dividend scheduled to be paid by the
   Issuer to holders of record of the Shares, in the period set out in Column headed Relevant Period will equal per Share the amount stated
   in respect of such Relevant Period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionFeatures">
  <xsd:sequence>
    <xsd:element name="asian" type=" Asian " minOccurs="0"/>
    <xsd:element name="barrier" type=" Barrier " minOccurs="0"/>
    <xsd:element name="knock" type=" Knock " minOccurs="0"/>
    <xsd:element name="passThrough" type=" PassThrough " minOccurs="0"/>
    <xsd:element name="dividendAdjustment" type=" DividendAdjustment " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrincipalExchangeAmount

[Table of contents]

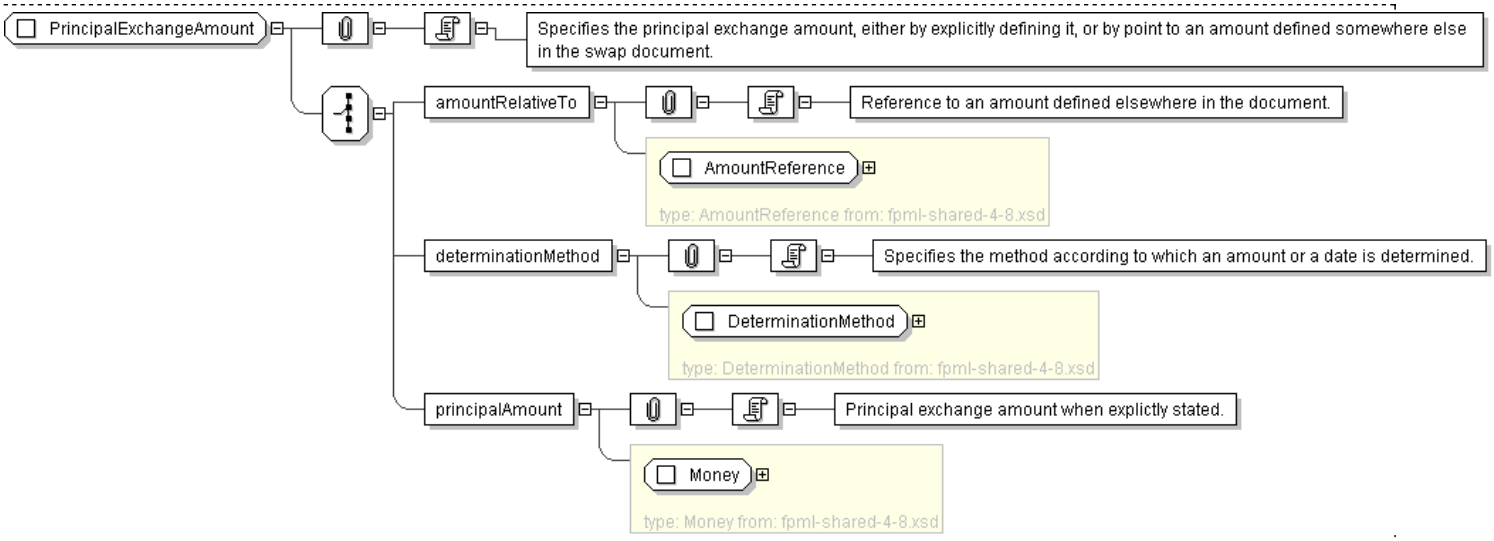
Super-types:	None
Sub-types:	None

Name	PrincipalExchangeAmount
Used by (from the same schema document)	Complex Type PrincipalExchangeDescriptions
Abstract	no
Documentation	Specifies the principal exchange amount, either by explicitly defining it, or by point to an amount defined somewhere else in the swap document.

XML Instance Representation

```
<...>
  Start Choice [1]
    <amountRelativeTo> AmountReference </amountRelativeTo> [1]
     'Reference to an amount defined elsewhere in the document.'

    <determinationMethod> DeterminationMethod </determinationMethod> [1]
     'Specifies the method according to which an amount or a date is determined.'

    <principalAmount> Money </principalAmount> [1]
     'Principal exchange amount when explictly stated.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeAmount">
  <xsd:choice>
    <xsd:element name="amountRelativeTo" type=" AmountReference "/>
    <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
    <xsd:element name="principalAmount" type=" Money "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrincipalExchangeDescriptions

[Table of contents]

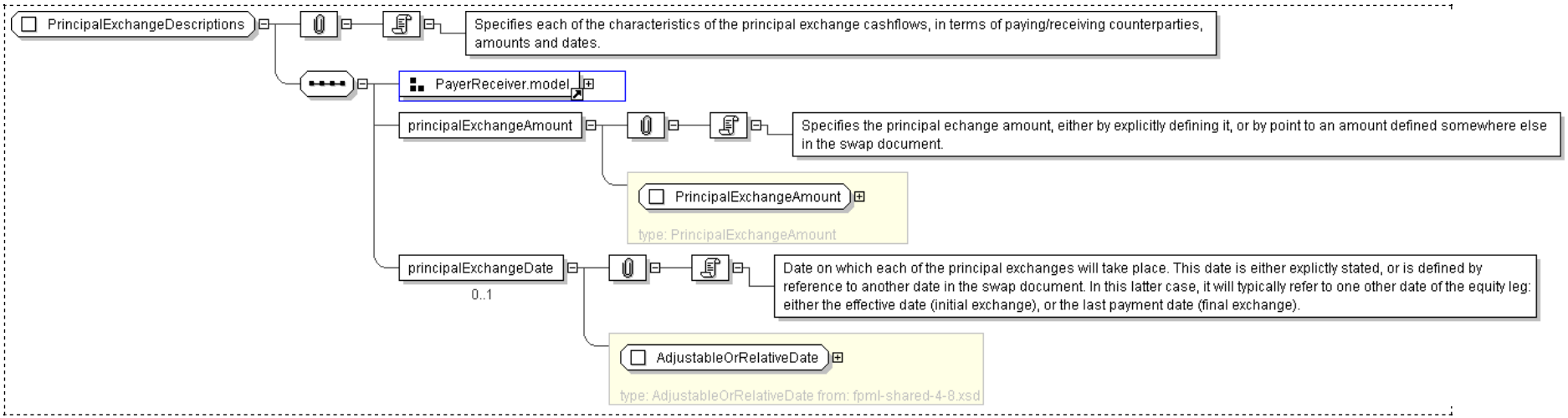
Super-types:	None
Sub-types:	None

Name	PrincipalExchangeDescriptions
Used by (from the same schema document)	Complex Type PrincipalExchangeFeatures
Abstract	no
Documentation	Specifies each of the characteristics of the principal exchange cashflows, in terms of paying/receiving counterparties, amounts and dates.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <principalExchangeAmount> PrincipalExchangeAmount </principalExchangeAmount> [1]
   'Specifies the principal echange amount, either by explicitly defining it, or by point to an amount defined somewhere else in the swap document.'

   <principalExchangeDate> AdjustableOrRelativeDate </principalExchangeDate> [0..1]
   'Date on which each of the principal exchanges will take place. This date is either explictly stated, or is defined by reference to another date in
   the swap document. In this latter case, it will typically refer to one other date of the equity leg: either the effective date (initial exchange), or
   the last payment date (final exchange).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeDescriptions">
   <xsd:sequence>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="principalExchangeAmount" type=" PrincipalExchangeAmount "/>
      <xsd:element name="principalExchangeDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrincipalExchangeFeatures

[Table of contents]

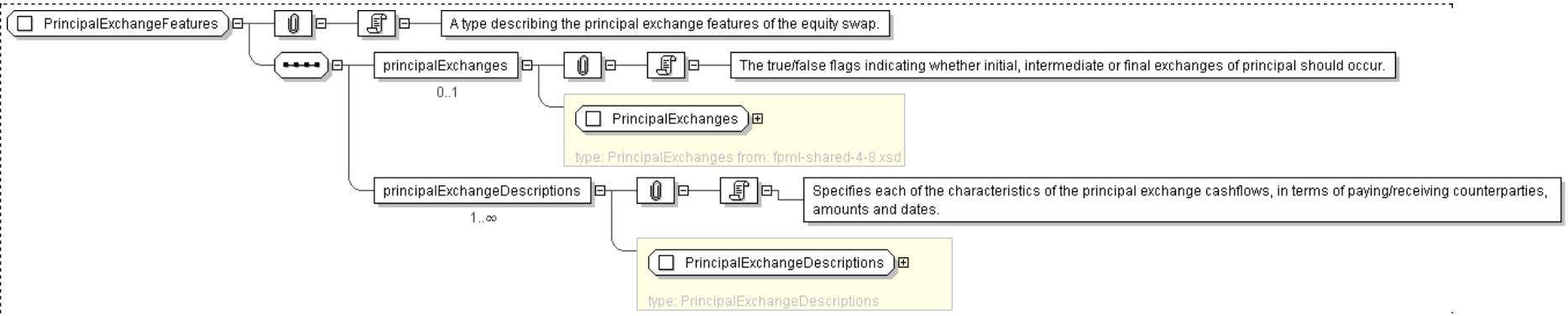
Super-types:	None
Sub-types:	None

Name	PrincipalExchangeFeatures
Used by (from the same schema document)	Complex Type ReturnSwapBase
Abstract	no
Documentation	A type describing the principal exchange features of the equity swap.

XML Instance Representation

```
<...>
  <principalExchanges> PrincipalExchanges </principalExchanges> [0..1]
  'The true/false flags indicating whether initial, intermediate or final exchanges of principal should occur.'

  <principalExchangeDescriptions> PrincipalExchangeDescriptions </principalExchangeDescriptions> [1..*]
  'Specifies each of the characteristics of the principal exchange cashflows, in terms of paying/receiving counterparties, amounts and dates.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeFeatures">
  <xsd:sequence>
    <xsd:element name="principalExchanges" type=" PrincipalExchanges " minOccurs="0"/>
    <xsd:element name="principalExchangeDescriptions" type=" PrincipalExchangeDescriptions " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Representations

[Table of contents]

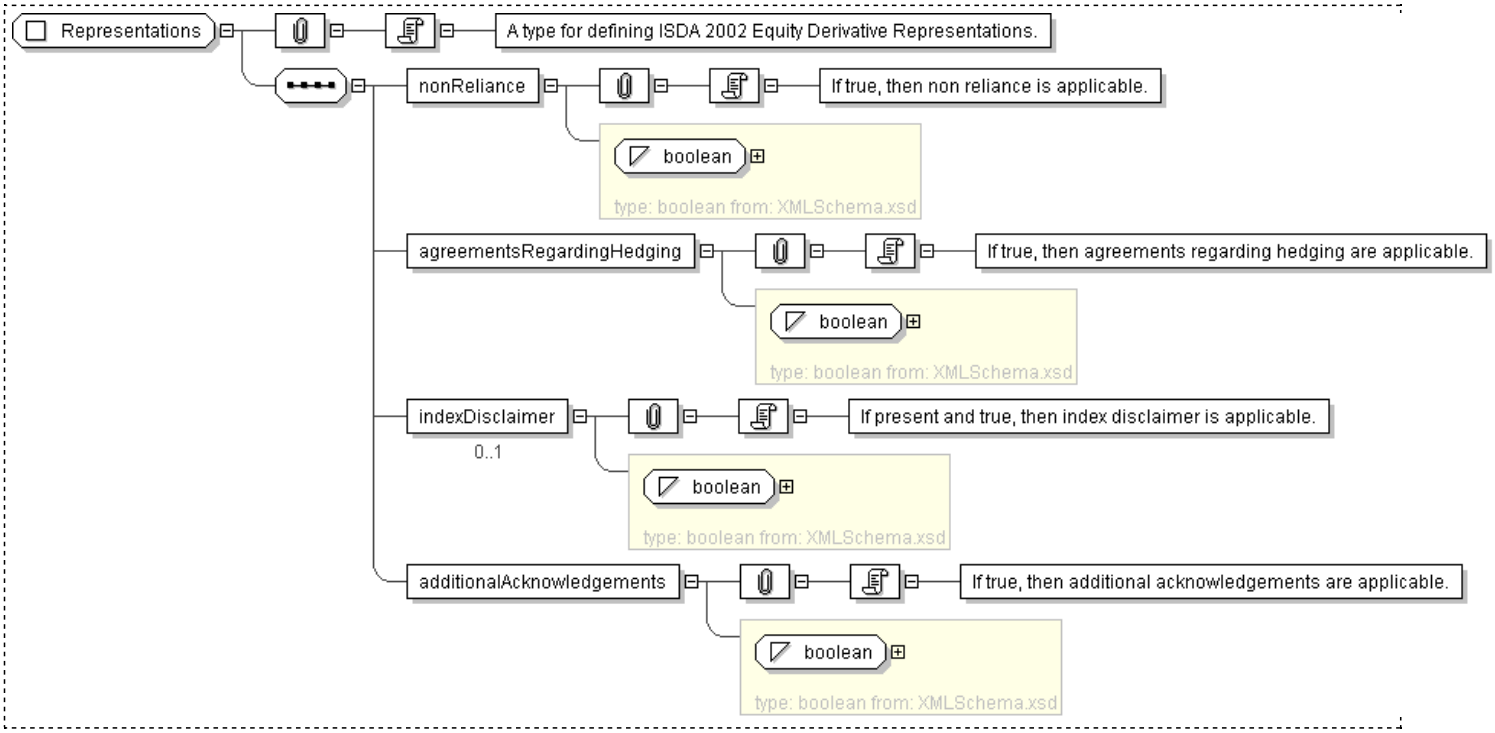
Super-types:	None
Sub-types:	None

Name	Representations
Used by (from the same schema document)	Complex Type ExtraordinaryEvents
Abstract	no
Documentation	A type for defining ISDA 2002 Equity Derivative Representations.

XML Instance Representation

```
<...>
   <nonReliance> xsd:boolean </nonReliance> [1]
   'If true, then non reliance is applicable.'

   <agreementsRegardingHedging> xsd:boolean </agreementsRegardingHedging> [1]
   'If true, then agreements regarding hedging are applicable.'

   <indexDisclaimer> xsd:boolean </indexDisclaimer> [0..1]
   'If present and true, then index disclaimer is applicable.'

   <additionalAcknowledgements> xsd:boolean </additionalAcknowledgements> [1]
   'If true, then additional acknowledgements are applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Representations">
   <xsd:sequence>
      <xsd:element name="nonReliance" type=" xsd:boolean "/>
      <xsd:element name="agreementsRegardingHedging" type=" xsd:boolean "/>
      <xsd:element name="indexDisclaimer" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="additionalAcknowledgements" type=" xsd:boolean "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Return

[Table of contents]

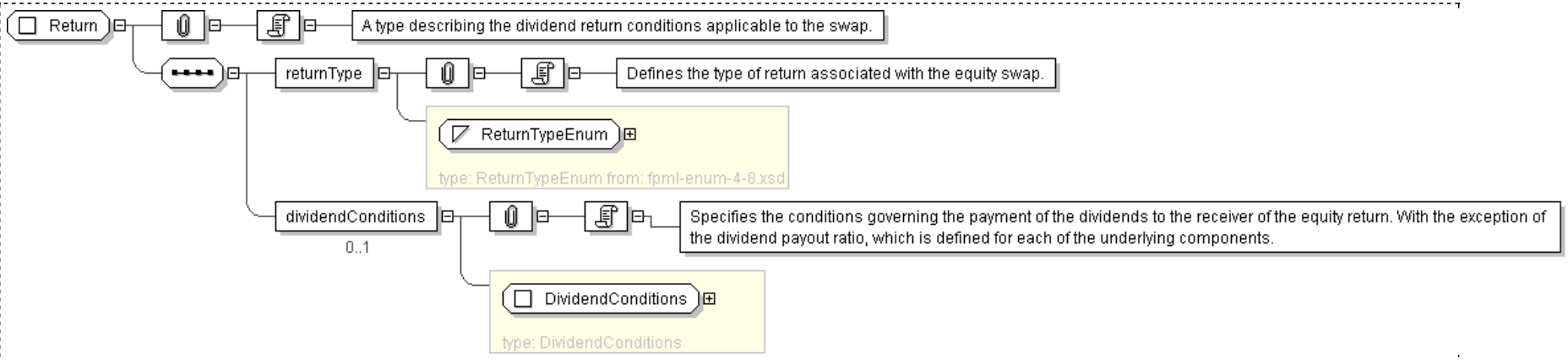
Super-types:	None
Sub-types:	None

Name	Return
Used by (from the same schema document)	Complex Type ReturnLeg
Abstract	no
Documentation	A type describing the dividend return conditions applicable to the swap.

XML Instance Representation

```
<...>
   <returnType> ReturnTypeEnum </returnType> [1]
   'Defines the type of return associated with the equity swap.'

   <dividendConditions> DividendConditions </dividendConditions> [0..1]
   'Specifies the conditions governing the payment of the dividends to the receiver of the equity return. With the exception of the
   dividend payout ratio, which is defined for each of the underlying components.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Return">
   <xsd:sequence>
      <xsd:element name="returnType" type=" ReturnTypeEnum "/>
      <xsd:element name="dividendConditions" type=" DividendConditions " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnLeg

[Table of contents]

Super-types:	Leg < ReturnSwapLeg (by extension) < ReturnSwapLegUnderlyer (by extension) < ReturnLeg (by extension)
Sub-types:	None

Name	ReturnLeg
Used by (from the same schema document)	Element returnLeg
Abstract	no
Documentation	A type describing the return leg of a return type swap.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version.
   Frequency at which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of this leg of the swap. When defined in relation
   to a date specified somewhere else in the document (through the relativeDate
   component), this element will typically point to the effective date of the
   other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of this leg of the swap. When defined in
   relation to a date specified somewhere else in the document (through the
   relativeDate component), this element will typically point to the termination
   date of the other leg of the swap.'

   <strikeDate> AdjustableOrRelativeDate </strikeDate> [0..1]
   'Specifies the strike date of this leg of the swap, used for forward starting
   swaps. When defined in relation to a date specified somewhere else in the
   document (through the relativeDate component), this element will typically by
   relative to the trade date of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component of the leg, which can be either one or many
   and consists in either equity, index or convertible bond component, or a
   combination of these.'

   <rateOfReturn> ReturnLegValuation </rateOfReturn> [1]
   'Element named \"valuation\" in versions prior to FpML 4.2 Second Working
   Draft. Specifies the terms of the initial price of the return type swap and of
   the subsequent valuations of the underlyer.'
```

<notional> ReturnSwapNotional **</notional>** [1]

'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine the actual amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest leg, the definition will typically point to the definition of the equity leg.'

<amount> ReturnSwapAmount **</amount>** [1]

'Element named \"equityAmount\" in versions prior to FpML 4.2 Second Working Draft. Specifies, in relation to each Payment Date, the amount to which the Payment Date relates. For equity swaps this element is equivalent to the Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.'

<return> Return **</return>** [1]

'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the amounts.'

<notionalAdjustments> NotionalAdjustmentEnum **</notionalAdjustments>** [1]

'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

<fxFeature> FxFeature **</fxFeature>** [0..1]

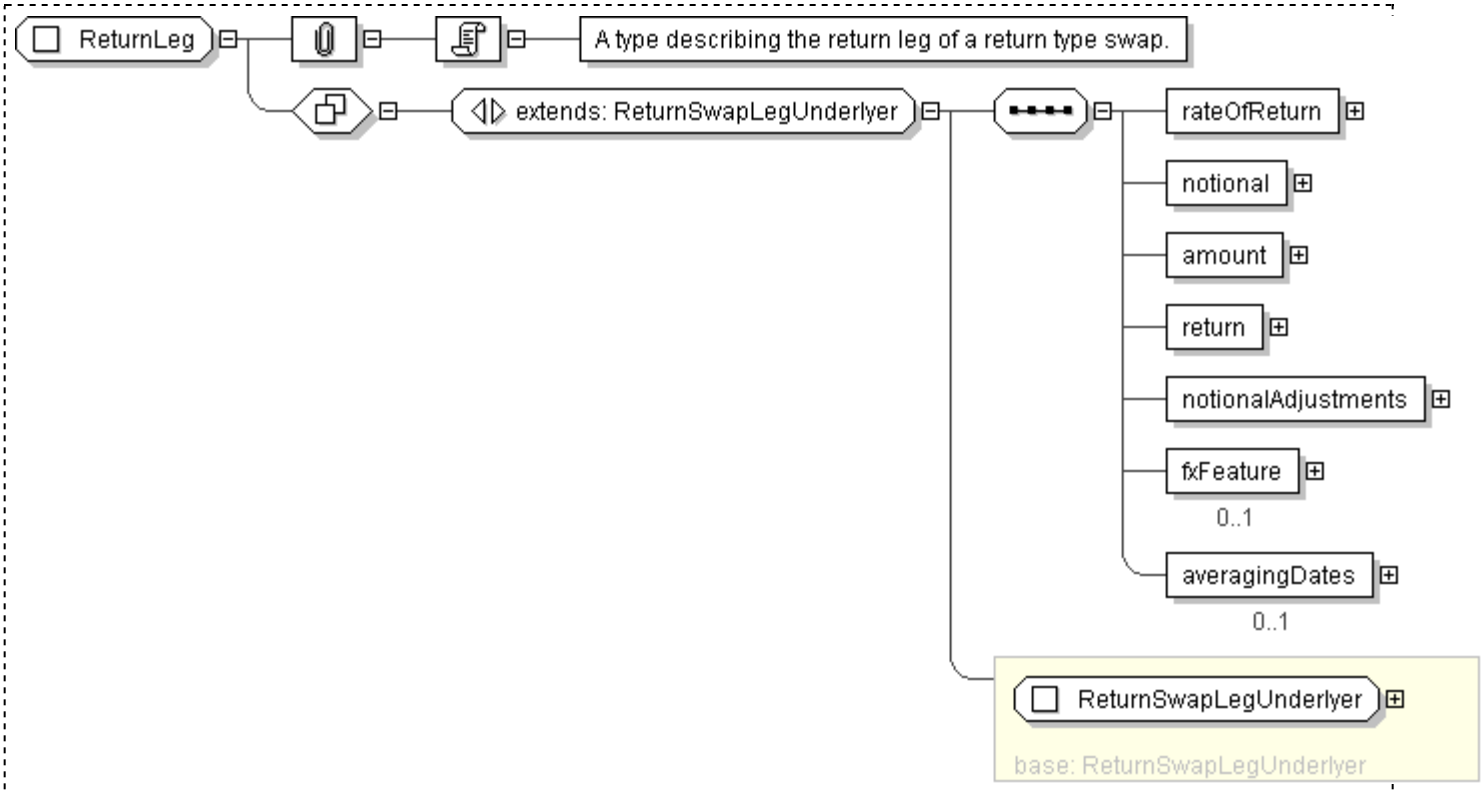
'A quanto or composite FX feature.'

<averagingDates> AveragingPeriod **</averagingDates>** [0..1]

'Averaging Dates used in the swap.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnLeg">
  <xsd:complexContent>
    <xsd:extension base=" ReturnSwapLegUnderlyer ">
      <xsd:sequence>
        <xsd:element name="rateOfReturn" type=" ReturnLegValuation "/>
        <xsd:element name="notional" type=" ReturnSwapNotional "/>
        <xsd:element name="amount" type=" ReturnSwapAmount "/>
        <xsd:element name="return" type=" Return "/>
```

```
            <xsd:element name="notionalAdjustments" type=" NotionalAdjustmentEnum "/>
            <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
            <xsd:element name="averagingDates" type=" AveragingPeriod " minOccurs="0"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnLegValuation

[Table of contents]

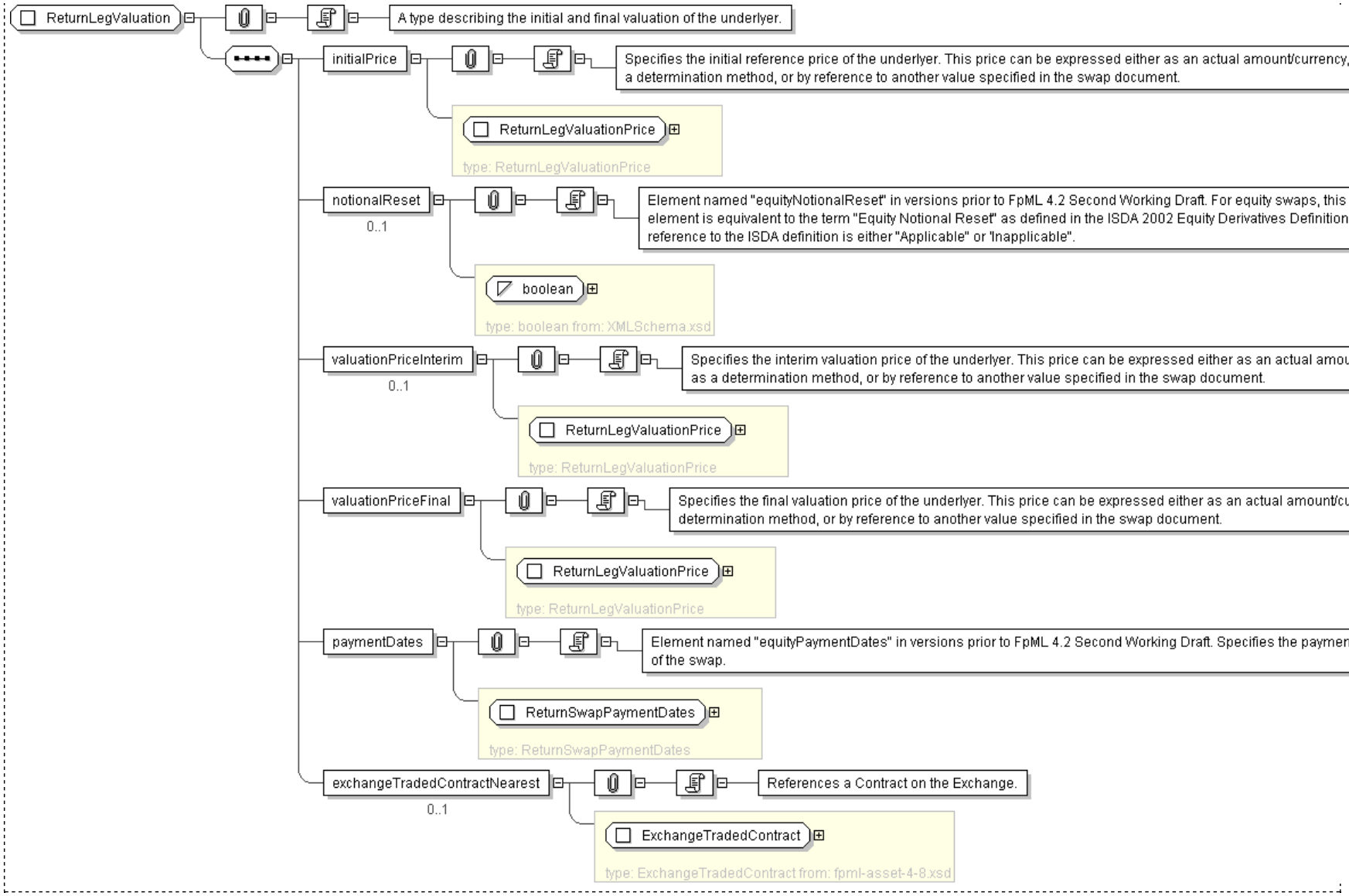
Super-types:	None
Sub-types:	None

Name	ReturnLegValuation
Used by (from the same schema document)	Complex Type ReturnLeg
Abstract	no
Documentation	A type describing the initial and final valuation of the underlyer.

XML Instance Representation

```
<...>
   <initialPrice> ReturnLegValuationPrice </initialPrice> [1]
   'Specifies the initial reference price of the underlyer. This price can be expressed either as an actual amount/currency, as a determination
   method, or by reference to another value specified in the swap document.'

   <notionalReset> xsd:boolean </notionalReset> [0..1]
   'Element named \"equityNotionalReset\" in versions prior to FpML 4.2 Second Working Draft. For equity swaps, this element is equivalent to
   the term \"Equity Notional Reset\" as defined in the ISDA 2002 Equity Derivatives Definitions. The reference to the ISDA definition is
   either \"Applicable\" or \'Inapplicable\".'

   <valuationPriceInterim> ReturnLegValuationPrice </valuationPriceInterim> [0..1]
   'Specifies the interim valuation price of the underlyer. This price can be expressed either as an actual amount/currency, as a determination
   method, or by reference to another value specified in the swap document.'

   <valuationPriceFinal> ReturnLegValuationPrice </valuationPriceFinal> [1]
   'Specifies the final valuation price of the underlyer. This price can be expressed either as an actual amount/currency, as a determination
   method, or by reference to another value specified in the swap document.'

   <paymentDates> ReturnSwapPaymentDates </paymentDates> [1]
   'Element named \"equityPaymentDates\" in versions prior to FpML 4.2 Second Working Draft. Specifies the payment dates of the swap.'

   <exchangeTradedContractNearest> ExchangeTradedContract </exchangeTradedContractNearest> [0..1]
   'References a Contract on the Exchange.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnLegValuation">
  <xsd:sequence>
    <xsd:element name="initialPrice" type=" ReturnLegValuationPrice "/>
    <xsd:element name="notionalReset" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="valuationPriceInterim" type=" ReturnLegValuationPrice " minOccurs="0"/>
    <xsd:element name="valuationPriceFinal" type=" ReturnLegValuationPrice "/>
    <xsd:element name="paymentDates" type=" ReturnSwapPaymentDates "/>
    <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnLegValuationPrice

[Table of contents]

Super-types:	Price < ReturnLegValuationPrice (by extension)
Sub-types:	None

Name	ReturnLegValuationPrice
Used by (from the same schema document)	Complex Type ReturnLegValuation , Complex Type ReturnLegValuation , Complex Type ReturnLegValuation
Abstract	no

XML Instance Representation

```
<...>
  <commission> Commission </commission> [0..1]
   'This optional component specifies the commission to be charged for executing
   the hedge transactions.'

  Start Choice [1]
    <determinationMethod> DeterminationMethod </determinationMethod> [1]
     'Specifies the method according to which an amount or a date is determined.'

    Start Group: EquityPrice.model [0..1]
      <grossPrice> ActualPrice </grossPrice> [0..1]
       'Specifies the price of the underlyer, before commissions.'

      <netPrice> ActualPrice </netPrice> [1]
       'Specifies the price of the underlyer, net of commissions.'

      <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
       'Specifies the accrued interest that are part of the dirty price in the
       case of a fixed income security or a convertible bond. Expressed in
       percentage of the notional.'

      <fxConversion> FxConversion </fxConversion> [0..1]
       'Specifies the currency conversion rate that applies to an amount. This
       rate can either be defined elsewhere in the document (case of a quanto
       swap), or explicitly described through this component.'

    End Group: EquityPrice.model
    <amountRelativeTo> AmountReference </amountRelativeTo> [1]
     'The href attribute value will be a pointer style reference to the element
     or component elsewhere in the document where the anchor amount is defined.'

    <grossPrice> ActualPrice </grossPrice> [0..1]
     'Specifies the price of the underlyer, before commissions.'

    <netPrice> ActualPrice </netPrice> [1]
     'Specifies the price of the underlyer, net of commissions.'

    <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
     'Specifies the accrued interest that are part of the dirty price in the case
     of a fixed income security or a convertible bond. Expressed in percentage of
     the notional.'

    <fxConversion> FxConversion </fxConversion> [0..1]
     'Specifies the currency conversion rate that applies to an amount. This rate
     can either be defined elsewhere in the document (case of a quanto swap), or
     explicitly described through this component.'

  End Choice
```

<cleanNetPrice> xsd:decimal </cleanNetPrice> [0..1]

'The net price excluding accrued interest. The \"Dirty Price\" for bonds is put in the \"netPrice\" element, which includes accrued interest. Thus netPrice - cleanNetPrice = accruedInterest. The currency and price expression for this field are the same as those for the (dirty) netPrice.'

<quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..1]

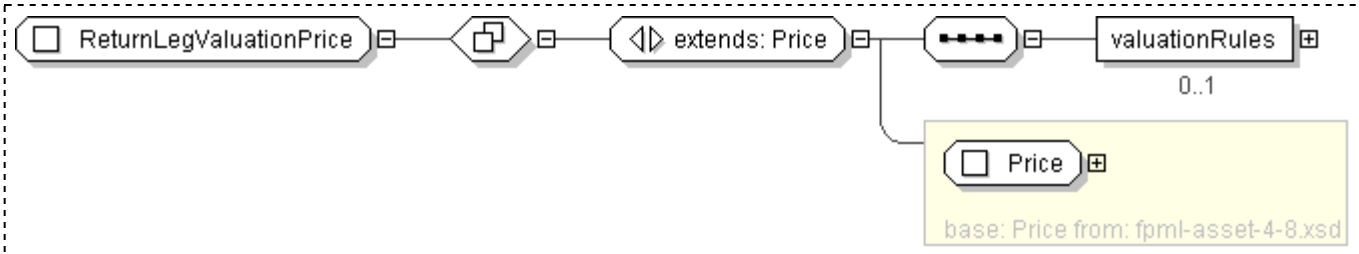
'Allows information about how the price was quoted to be provided.'

<valuationRules> EquityValuation </valuationRules> [0..1]

'Element named \"equityValuation\" in versions prior to FpML 4.2 Second Working Draft.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnLegValuationPrice">
   <xsd:complexContent>
      <xsd:extension base=" Price ">
         <xsd:sequence>
            <xsd:element name="valuationRules" type=" EquityValuation " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwap

[Table of contents]

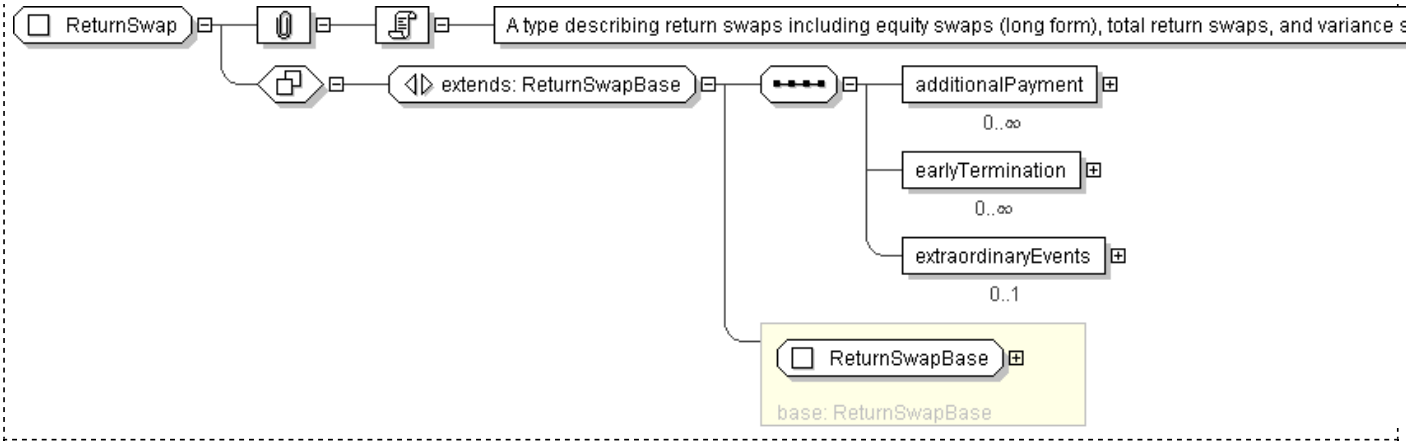
Super-types:	Product < ReturnSwapBase (by extension) < ReturnSwap (by extension)
Sub-types:	None

Name	ReturnSwap
Used by (from the same schema document)	Element returnSwap
Abstract	no
Documentation	A type describing return swaps including equity swaps (long form), total return swaps, and variance swaps.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a
   coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not strictly an
   enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation
where an implementor wishes to indicate who has manufactured the Swap through representing them as
the Seller. It may be removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives
      the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this
      the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights
      defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions
      Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   End Group: BuyerSeller.model
   <returnSwapLeg> ... </returnSwapLeg> [1..*]
   <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
   'This is used to document a Fully Funded Return Swap.'

   <additionalPayment> ReturnSwapAdditionalPayment </additionalPayment> [0..*]
   'Specifies additional payment(s) between the principal parties to the trade. This component extends
   some of the features of the additionalPayment component developed by the FpML industry group.
   Appropriate discussions will determine whether it would be appropriate to extend the shared
   component in order to meet the further requirements of equity swaps.'

   <earlyTermination> ReturnSwapEarlyTermination </earlyTermination> [0..*]
   'Specifies, for one or for both the parties to the trade, the date from which it can early
   terminate it.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
   'Where the underlying is shares, specifies events affecting the issuer of those shares that may
   require the terms of the transaction to be adjusted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwap">
  <xsd:complexContent>
    <xsd:extension base=" ReturnSwapBase ">
      <xsd:sequence>
        <xsd:element name="additionalPayment" type=" ReturnSwapAdditionalPayment " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="earlyTermination" type=" ReturnSwapEarlyTermination " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapAdditionalPayment

[Table of contents]

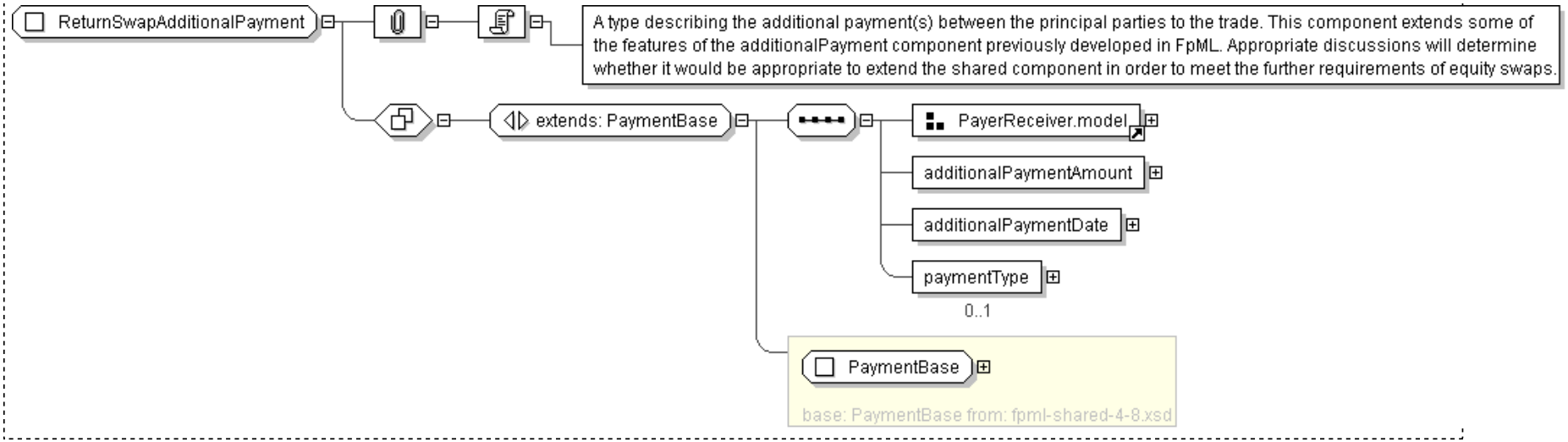
Super-types:	PaymentBase < ReturnSwapAdditionalPayment (by extension)
Sub-types:	None

Name	ReturnSwapAdditionalPayment
Used by (from the same schema document)	Complex Type ReturnSwap
Abstract	no
Documentation	A type describing the additional payment(s) between the principal parties to the trade. This component extends some of the features of the additionalPayment component previously developed in FpML. Appropriate discussions will determine whether it would be appropriate to extend the shared component in order to meet the further requirements of equity swaps.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <additionalPaymentAmount> AdditionalPaymentAmount </additionalPaymentAmount> [1]
   'Specifies the amount of the fee along with, when applicable, the formula that supports its determination.'

   <additionalPaymentDate> AdjustableOrRelativeDate </additionalPaymentDate> [1]
   'Specifies the value date of the fee payment/receipt.'

   <paymentType> PaymentType </paymentType> [0..1]
   'Classification of the payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapAdditionalPayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="additionalPaymentAmount" type=" AdditionalPaymentAmount "/>
        <xsd:element name="additionalPaymentDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="paymentType" type=" PaymentType " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapAmount

[Table of contents]

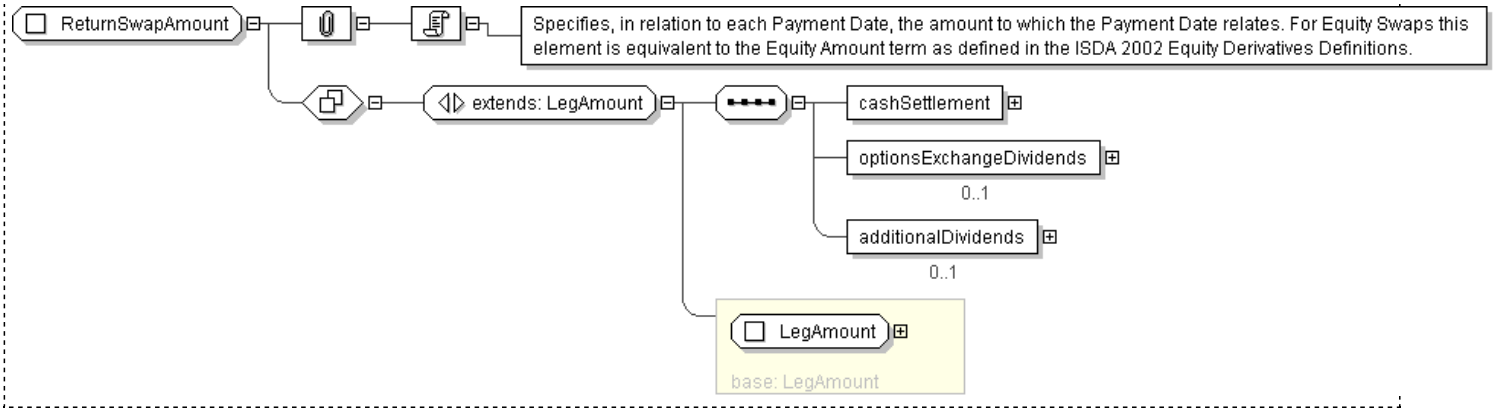
Super-types:	LegAmount < ReturnSwapAmount (by extension)
Sub-types:	• DeprecatedVarianceAmount (by extension)

Name	ReturnSwapAmount
Used by (from the same schema document)	Complex Type ReturnLeg
Abstract	no
Documentation	Specifies, in relation to each Payment Date, the amount to which the Payment Date relates. For Equity Swaps this element is equivalent to the Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.

XML Instance Representation

```
<...>
  Start Group: CurrencyAndDeterminationMethod.model [0..1]
    Start Choice [1]
      <currency> Currency </currency> [1]
      'The currency in which an amount is denominated.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <currencyReference> IdentifiedCurrencyReference </currencyReference> [1]
      'Reference to a currency defined elsewhere in the document'

    End Choice
  End Group: CurrencyAndDeterminationMethod.model
  <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
  'DEPRECATED. Currency in which the payment relating to the leg amount (equity amount or interest amount) or
  the dividend will be denominated.'

  Start Choice [1]
    <referenceAmount> ReferenceAmount </referenceAmount> [1]
    'Specifies the reference Amount when this term either corresponds to the standard ISDA Definition (either
    the 2002 Equity Definition for the Equity Amount, or the 2000 Definition for the Interest Amount), or
    points to a term defined elsewhere in the swap document.'

    <formula> Formula </formula> [1]
    'Specifies a formula, with its description and components.'

    <encodedDescription> xsd:base64Binary </encodedDescription> [1]
    'Description of the leg amount when represented through an encoded image.'

    <variance> DeprecatedVariance </variance> [1]
    'DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be
    used for Variance Swaps, use the Variance Swap Product. Specifies Variance for Variance Leg.'

  End Choice
  <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
  'Specifies the date on which a calculation or an observation will be performed for the purpose of defining
  the Equity Amount, and in accordance to the definition terms of this latter.'

  <cashSettlement> xsd:boolean </cashSettlement> [1]
  'If true, then cash settlement is applicable.'

  <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
  'If present and true, then options exchange dividends are applicable.'

  <additionalDividends> xsd:boolean </additionalDividends> [0..1]
  'If present and true, then additional dividends are applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapAmount">
  <xsd:complexContent>
    <xsd:extension base=" LegAmount ">
      <xsd:sequence>
        <xsd:element name="cashSettlement" type=" xsd:boolean "/>
        <xsd:element name="optionsExchangeDividends" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="additionalDividends" type=" xsd:boolean " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapBase

[Table of contents]

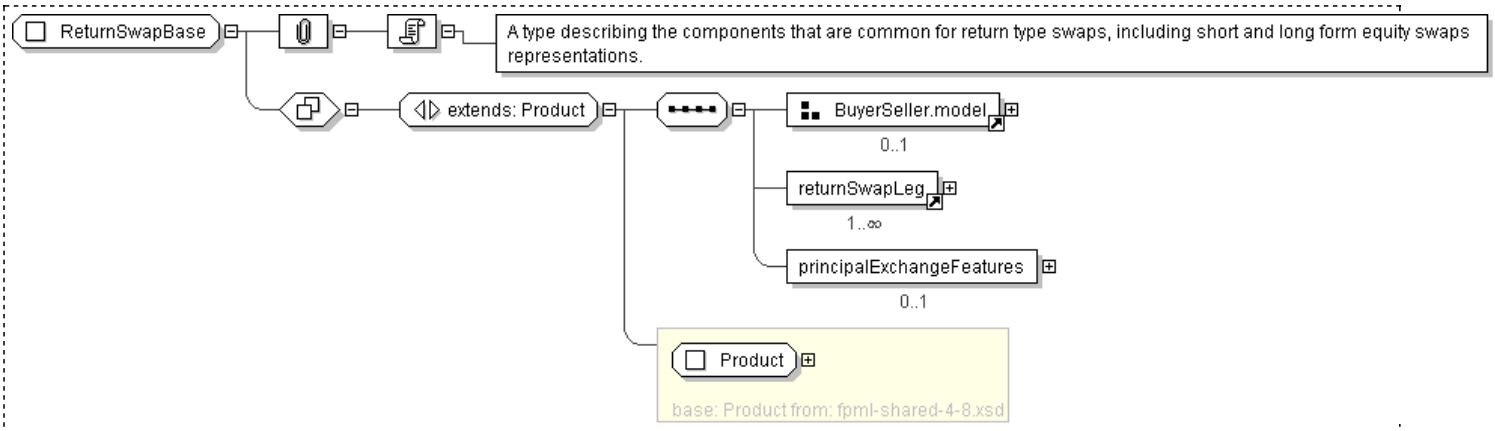
Super-types:	Product < ReturnSwapBase (by extension)
Sub-types:	• ReturnSwap (by extension)

Name	ReturnSwapBase
Abstract	yes
Documentation	A type describing the components that are common for return type swaps, including short and long form equity swaps representations.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where an
implementor wishes to indicate who has manufactured the Swap through representing them as the Seller. It may be
removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
      defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
      this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In
      the case of FRAs this is the floating rate payer.'

   End Group: BuyerSeller.model
   <returnSwapLeg> ... </returnSwapLeg> [1..*]
   <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
   'This is used to document a Fully Funded Return Swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapBase" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model " minOccurs="0"/>
        <xsd:element ref=" returnSwapLeg " maxOccurs="unbounded"/>
        <xsd:element name="principalExchangeFeatures" type=" PrincipalExchangeFeatures " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapEarlyTermination

[Table of contents]

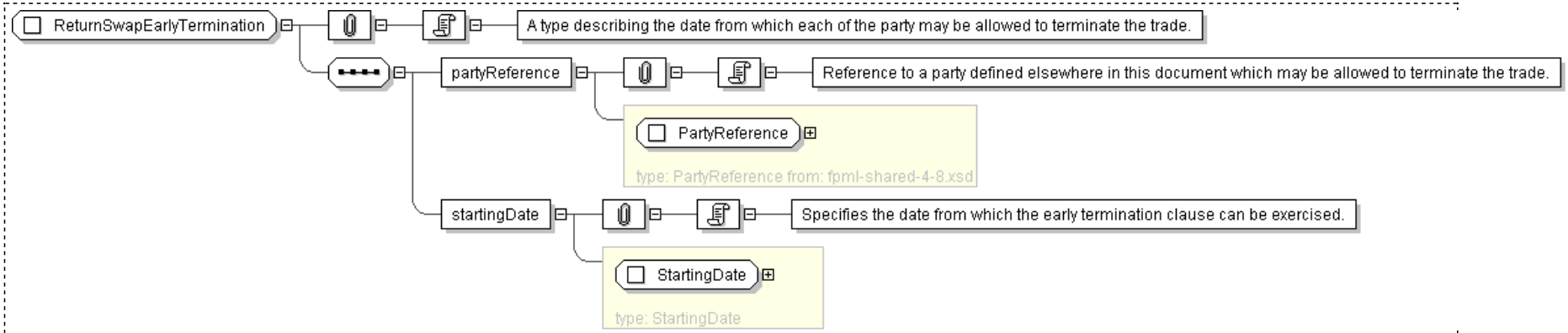
Super-types:	None
Sub-types:	None

Name	ReturnSwapEarlyTermination
Used by (from the same schema document)	Complex Type ReturnSwap
Abstract	no
Documentation	A type describing the date from which each of the party may be allowed to terminate the trade.

XML Instance Representation

```
<...>
   <partyReference> PartyReference </partyReference> [1]
   'Reference to a party defined elsewhere in this document which may be allowed to terminate the trade.'

   <startingDate> StartingDate </startingDate> [1]
   'Specifies the date from which the early termination clause can be exercised.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapEarlyTermination">
   <xsd:sequence>
      <xsd:element name="partyReference" type=" PartyReference "/>
      <xsd:element name="startingDate" type=" StartingDate "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapLeg

[Table of contents]

Super-types: Leg < ReturnSwapLeg (by extension)

Sub-types:

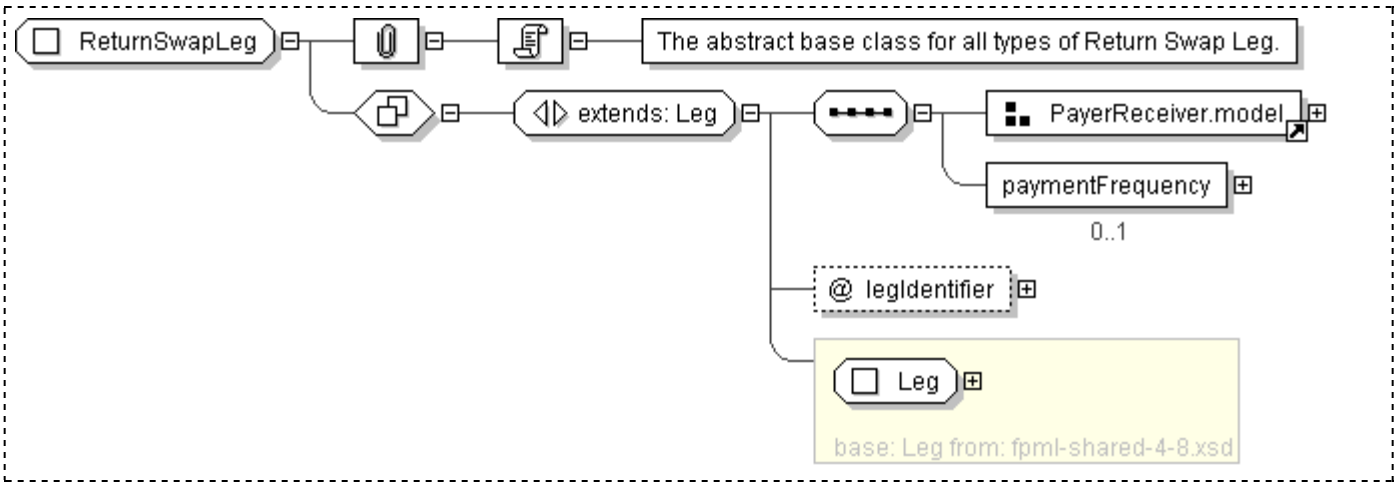
- DeprecatedVarianceLeg (by extension)
- InterestLeg (by extension)
- ReturnSwapLegUnderlyer (by extension)
  - ReturnLeg (by extension)

Name	ReturnSwapLeg
Used by (from the same schema document)	Element returnSwapLeg
Abstract	yes
Documentation	The abstract base class for all types of Return Swap Leg.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version.
   Frequency at which this leg pays.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapLeg" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="paymentFrequency" type=" Frequency " minOccurs="0"
        deprecated="true" deprecatedReason="Payment Frequency is controlled by the
        implementations of this abstract base class"/>
      </xsd:sequence>
      <xsd:attribute name="legIdentifier" type=" xsd:ID " deprecated="true"
      deprecatedReason="All attributes of type=xsd:ID should have name=id following
      FpML Architecture"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapLegUnderlyer

[Table of contents]

Super-types: Leg < ReturnSwapLeg (by extension) < ReturnSwapLegUnderlyer (by extension)

Sub-types:

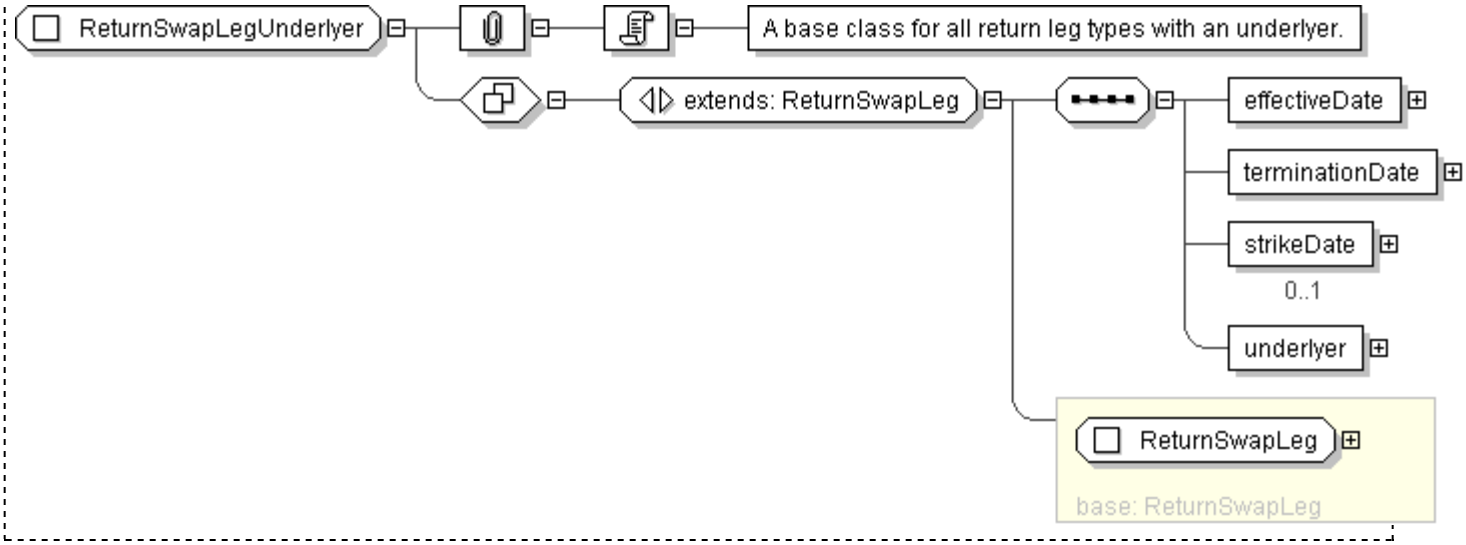
- ReturnLeg (by extension)

Name	ReturnSwapLegUnderlyer
Abstract	yes
Documentation	A base class for all return leg types with an underlyer.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version.
   Frequency at which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of this leg of the swap. When defined in relation
   to a date specified somewhere else in the document (through the relativeDate
   component), this element will typically point to the effective date of the
   other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of this leg of the swap. When defined in
   relation to a date specified somewhere else in the document (through the
   relativeDate component), this element will typically point to the termination
   date of the other leg of the swap.'

   <strikeDate> AdjustableOrRelativeDate </strikeDate> [0..1]
   'Specifies the strike date of this leg of the swap, used for forward starting
   swaps. When defined in relation to a date specified somewhere else in the
   document (through the relativeDate component), this element will typically by
   relative to the trade date of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component of the leg, which can be either one or many
   and consists in either equity, index or convertible bond component, or a
   combination of these.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapLegUnderlyer" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="strikeDate" type=" AdjustableOrRelativeDate "
            minOccurs="0"/>
            <xsd:element name="underlyer" type=" Underlyer "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapNotional

[Table of contents]

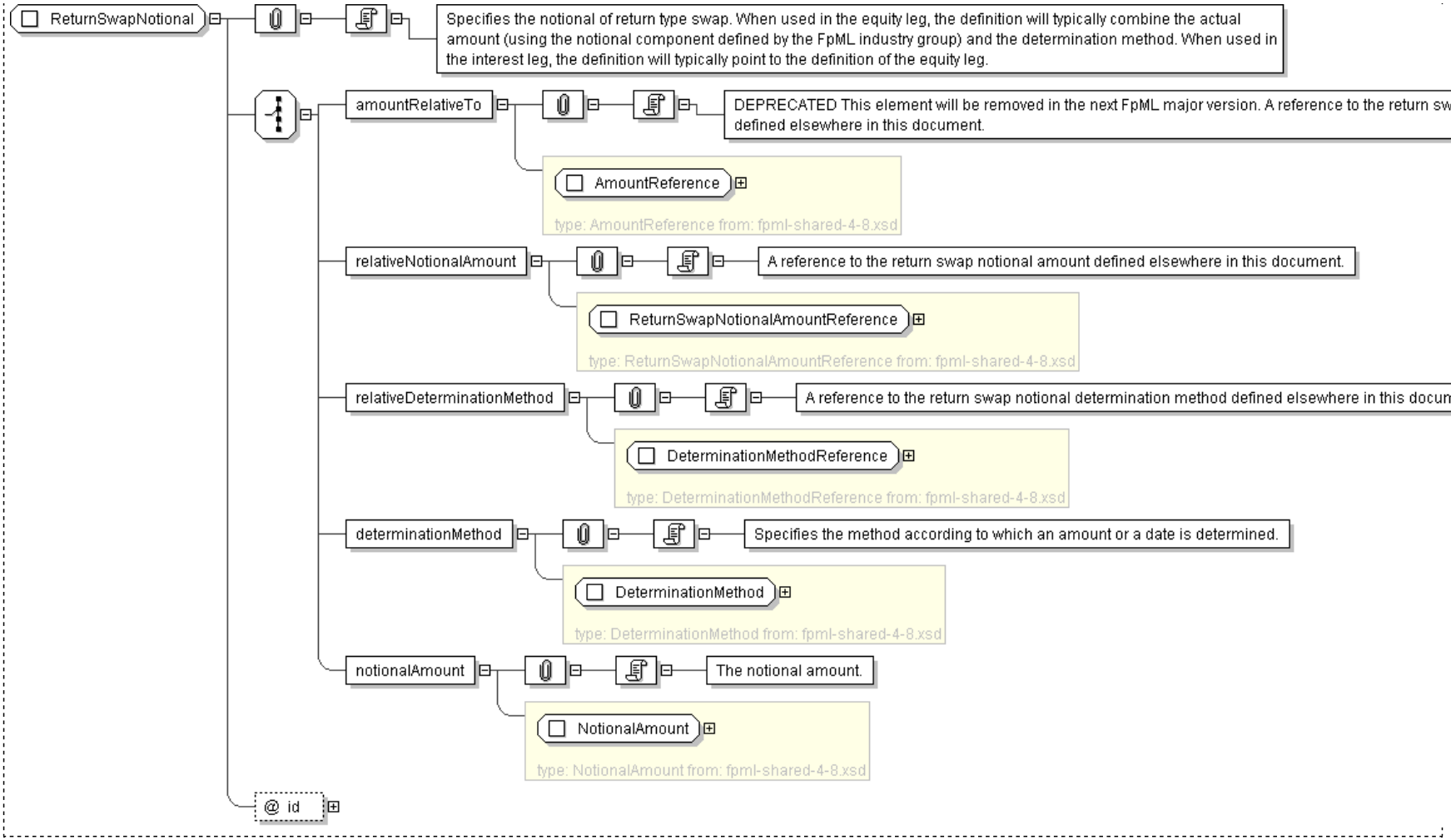
Super-types:	None
Sub-types:	None

Name	ReturnSwapNotional
Used by (from the same schema document)	Complex Type InterestLeg , Complex Type ReturnLeg
Abstract	no
Documentation	Specifies the notional of return type swap. When used in the equity leg, the definition will typically combine the actual amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest leg, the definition will typically point to the definition of the equity leg.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'DEPRECATED This element will be removed in the next FpML major version. A reference to the return swap notional defined elsewhere in
      this document.'

      <relativeNotionalAmount> ReturnSwapNotionalAmountReference </relativeNotionalAmount> [1]
      'A reference to the return swap notional amount defined elsewhere in this document.'

      <relativeDeterminationMethod> DeterminationMethodReference </relativeDeterminationMethod> [1]
      'A reference to the return swap notional determination method defined elsewhere in this document.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <notionalAmount> NotionalAmount </notionalAmount> [1]
      'The notional amount.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapNotional">
  <xsd:choice>
    <xsd:element name="amountRelativeTo" type=" AmountReference " deprecated="true" deprecatedReason="amountRelativeTo should not be used to
    the return swap notional, use the relativeNotionalAmount or relativeDeterminationMethod."/>
    <xsd:element name="relativeNotionalAmount" type=" ReturnSwapNotionalAmountReference "/>
    <xsd:element name="relativeDeterminationMethod" type=" DeterminationMethodReference "/>
    <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
    <xsd:element name="notionalAmount" type=" NotionalAmount "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapPaymentDates

[Table of contents]

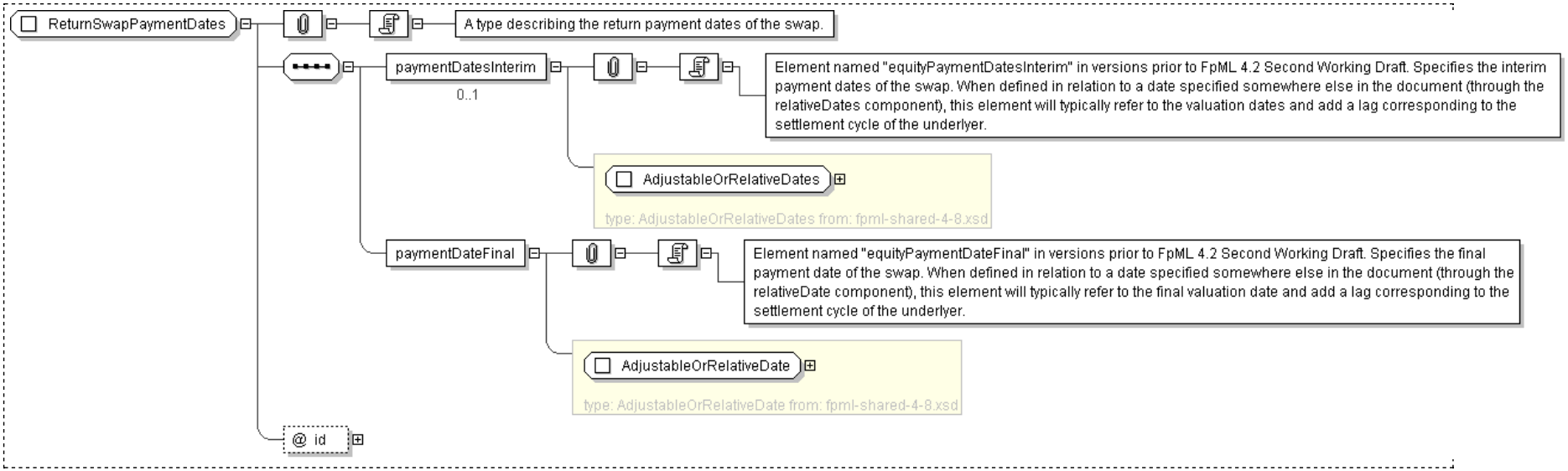
Super-types:	None
Sub-types:	None

Name	ReturnSwapPaymentDates
Used by (from the same schema document)	Complex Type ReturnLegValuation
Abstract	no
Documentation	A type describing the return payment dates of the swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <paymentDatesInterim> AdjustableOrRelativeDates </paymentDatesInterim> [0..1]
   'Element named \"equityPaymentDatesInterim\" in versions prior to FpML 4.2 Second Working Draft. Specifies the interim payment dates of the swap.
   When defined in relation to a date specified somewhere else in the document (through the relativeDates component), this element will typically
   refer to the valuation dates and add a lag corresponding to the settlement cycle of the underlyer.'

   <paymentDateFinal> AdjustableOrRelativeDate </paymentDateFinal> [1]
   'Element named \"equityPaymentDateFinal\" in versions prior to FpML 4.2 Second Working Draft. Specifies the final payment date of the swap. When
   defined in relation to a date specified somewhere else in the document (through the relativeDate component), this element will typically refer to
   the final valuation date and add a lag corresponding to the settlement cycle of the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapPaymentDates">
  <xsd:sequence>
    <xsd:element name="paymentDatesInterim" type=" AdjustableOrRelativeDates " minOccurs="0"/>
    <xsd:element name="paymentDateFinal" type=" AdjustableOrRelativeDate "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StartingDate

[Table of contents]

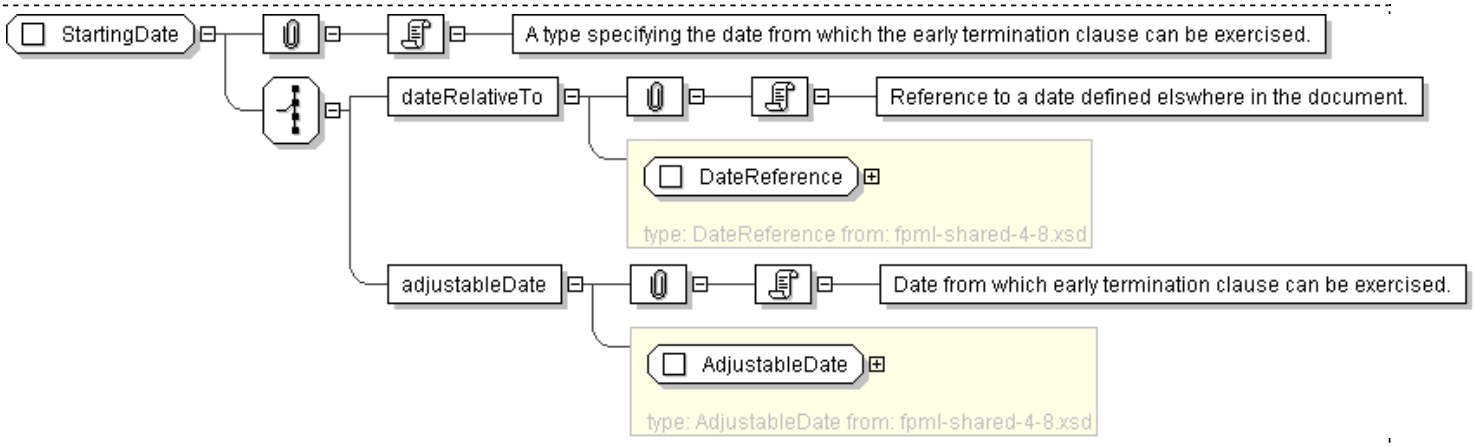
Super-types:	None
Sub-types:	None

Name	StartingDate
Used by (from the same schema document)	Complex Type DeprecatedVarianceAmount , Complex Type ReturnSwapEarlyTermination
Abstract	no
Documentation	A type specifying the date from which the early termination clause can be exercised.

XML Instance Representation

```
<...>
   Start Choice [1]
      <dateRelativeTo> DateReference </dateRelativeTo> [1]
      'Reference to a date defined elswhere in the document.'

      <adjustableDate> AdjustableDate </adjustableDate> [1]
      'Date from which early termination clause can be exercised.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StartingDate">
   <xsd:choice>
      <xsd:element name="dateRelativeTo" type=" DateReference "/>
      <xsd:element name="adjustableDate" type=" AdjustableDate "/>
   </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StubCalculationPeriod

[Table of contents]

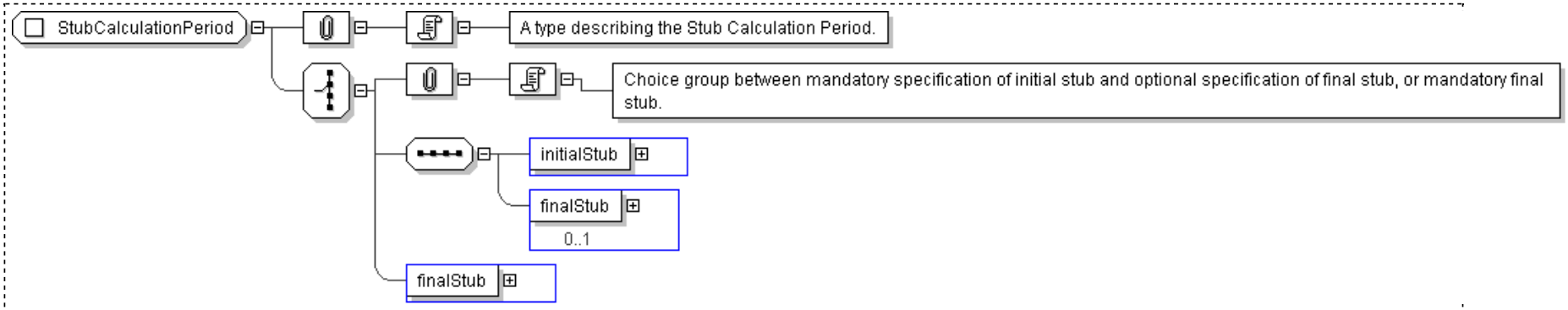
Super-types:	None
Sub-types:	None

Name	StubCalculationPeriod
Used by (from the same schema document)	Complex Type InterestLeg
Abstract	no
Documentation	A type describing the Stub Calculation Period.

XML Instance Representation

```
<...>
  Start Choice [1]
'Choice group between mandatory specification of initial stub and optional specification of final stub, or mandatory final stub.'


    <initialStub> Stub </initialStub> [1]
    <finalStub> Stub </finalStub> [0..1]
    <finalStub> Stub </finalStub> [1]
  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StubCalculationPeriod">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="initialStub" type=" Stub "/>
      <xsd:element name="finalStub" type=" Stub " minOccurs="0"/>
    </xsd:sequence>
    <xsd:element name="finalStub" type=" Stub "/>
  </xsd:choice>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Variance

[Table of contents]

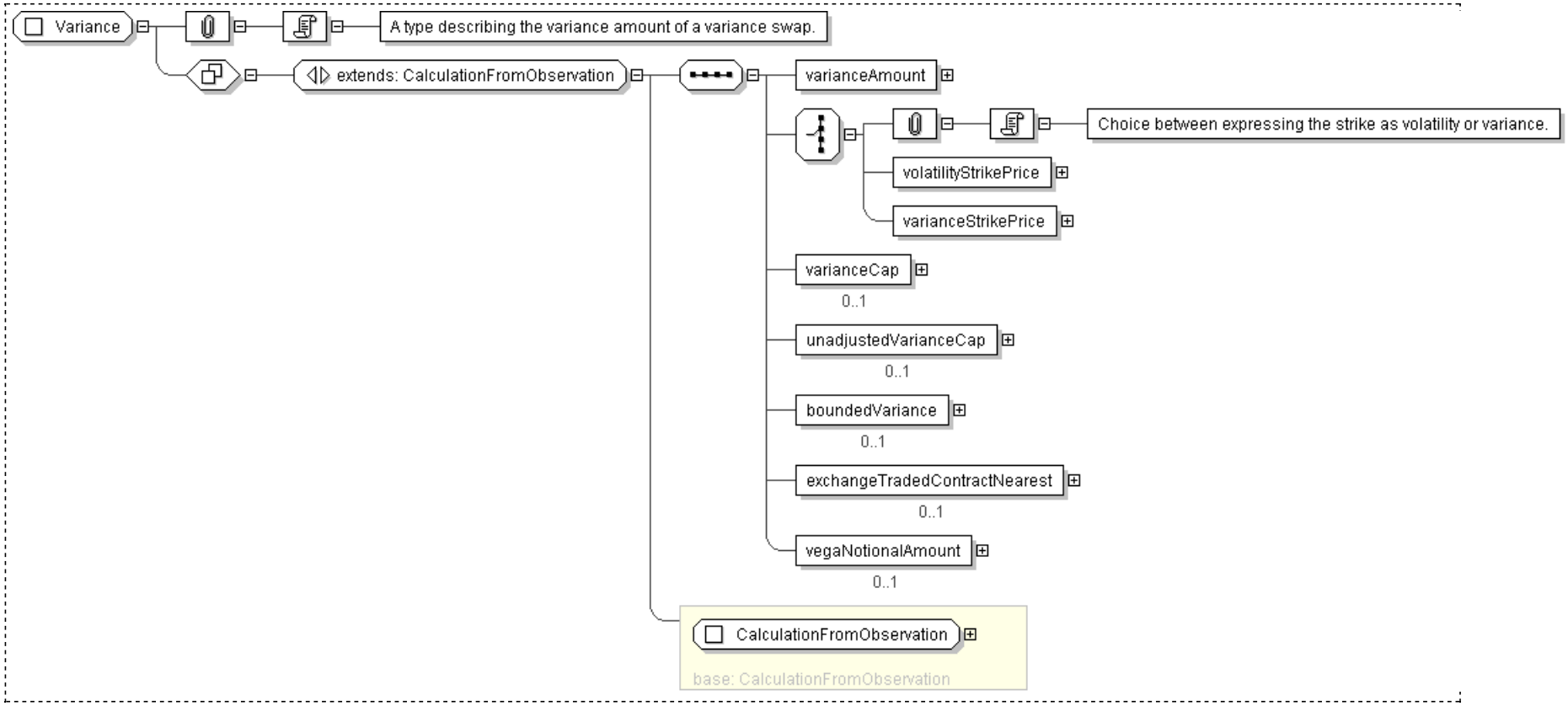
Super-types:	CalculationFromObservation < Variance (by extension)
Sub-types:	None

Name	Variance
Abstract	no
Documentation	A type describing the variance amount of a variance swap.

XML Instance Representation

```
<...>
   Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      'Contract will strike off this initial level.'

      <closingLevel> xsd:boolean </closingLevel> [1]
      'If true this contract will strike off the closing level of the default exchange traded contract.'

      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If true this contract will strike off the expiring level of the default exchange traded contract.'

   End Choice
   <expectedN> xsd:positiveInteger </expectedN> [0..1]
   'Expected number of trading days.'

   <varianceAmount> Money </varianceAmount> [1]
   'Variance amount, which is a cash multiplier.'

   Start Choice [1]
'Choice between expressing the strike as volatility or variance.'


      <volatilityStrikePrice> NonNegativeDecimal </volatilityStrikePrice> [1]
      <varianceStrikePrice> NonNegativeDecimal </varianceStrikePrice> [1]
   End Choice
   <varianceCap> xsd:boolean </varianceCap> [0..1]
   'If present and true, then variance cap is applicable.'

   <unadjustedVarianceCap> PositiveDecimal </unadjustedVarianceCap> [0..1]
   'For use when varianceCap is applicable. Contains the scaling factor of the Variance Cap that can differ on a trade-by-trade
   basis in the European market. For example, a Variance Cap of 2.5^2 x Variance Strike Price has an unadjustedVarianceCap of 2.5.'

   <boundedVariance> BoundedVariance </boundedVariance> [0..1]
   'Conditions which bound variance. The contract specifies one or more boundary levels. These levels are expressed as prices for
   confirmation purposes Underlyer price must be equal to or higher than Lower Barrier is known as Up Conditional Swap Underlyer
   price must be equal to or lower than Upper Barrier is known as Down Conditional Swap Underlyer price must be equal to or higher
   than Lower Barrier and must be equal to or lower than Upper Barrier is known as Barrier Conditional Swap.'

   <exchangeTradedContractNearest> ExchangeTradedContract </exchangeTradedContractNearest> [0..1]
   'Specification of the exchange traded contract nearest.'

   <vegaNotionalAmount> xsd:decimal </vegaNotionalAmount> [0..1]
   'Vega Notional represents the approximate gain/loss at maturity for a 1% difference between RVol (realised vol) and KVol (strike
   vol). It does not necessarily represent the Vega Risk of the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Variance">
  <xsd:complexContent>
    <xsd:extension base=" CalculationFromObservation ">
      <xsd:sequence>
        <xsd:element name="varianceAmount" type=" Money "/>
        <xsd:choice>
          <xsd:element name="volatilityStrikePrice" type=" NonNegativeDecimal "/>
          <xsd:element name="varianceStrikePrice" type=" NonNegativeDecimal "/>
        </xsd:choice>
        <xsd:element name="varianceCap" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="unadjustedVarianceCap" type=" PositiveDecimal " minOccurs="0"/>
        <xsd:element name="boundedVariance" type=" BoundedVariance " minOccurs="0"/>
        <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract " minOccurs="0"/>
        <xsd:element name="vegaNotionalAmount" type=" xsd:decimal " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: fxAverageRateOption
  - Element: fxBarrierOption
  - Element: fxDigitalOption
  - Element: fxSimpleOption
  - Element: fxSingleLeg
  - Element: fxSwap
  - Element: termDeposit
- Global Definitions
  - Complex Type: CutName
  - Complex Type: ExchangeRate
  - Complex Type: ExpiryDateTime
  - Complex Type: FxAmericanTrigger
  - Complex Type: FxAverageRateObservationDate
  - Complex Type: FxAverageRateObservationSchedule
  - Complex Type: FxAverageRateOption
  - Complex Type: FxBarrier
  - Complex Type: FxBarrierOption
  - Complex Type: FxDigitalOption
  - Complex Type: FxEuropeanTrigger
  - Complex Type: FxLeg
  - Complex Type: FxOptionLeg
  - Complex Type: FxOptionPayout
  - Complex Type: FxOptionPremium
  - Complex Type: FxStrikePrice
  - Complex Type: FxSwap
  - Complex Type: ObservedRates
  - Complex Type: PremiumQuote
  - Complex Type: QuotedAs
  - Complex Type: SideRate
  - Complex Type: SideRates
  - Complex Type: TermDeposit
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-shared-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's

content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The

`xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

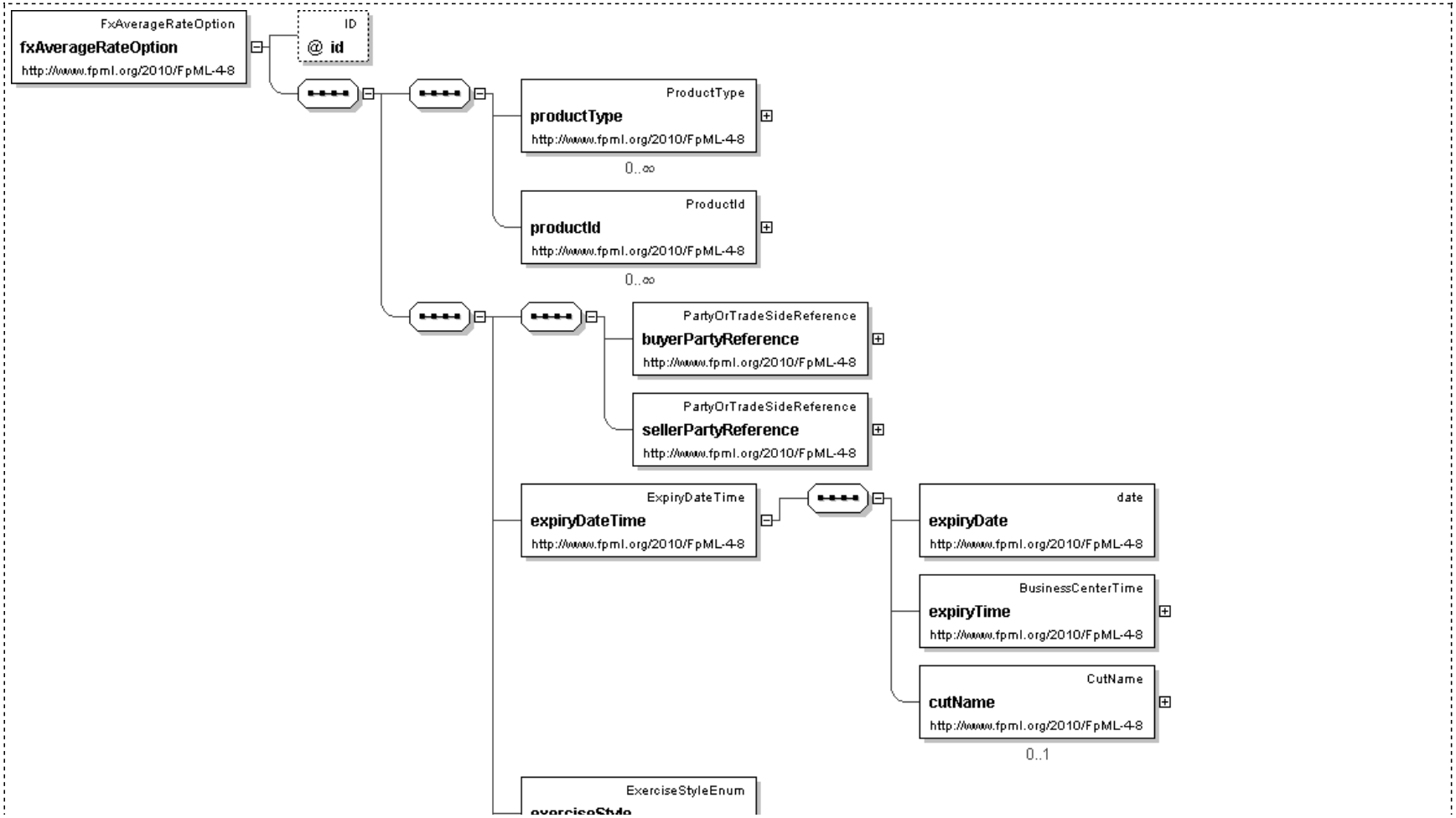
Element: fxAverageRateOption

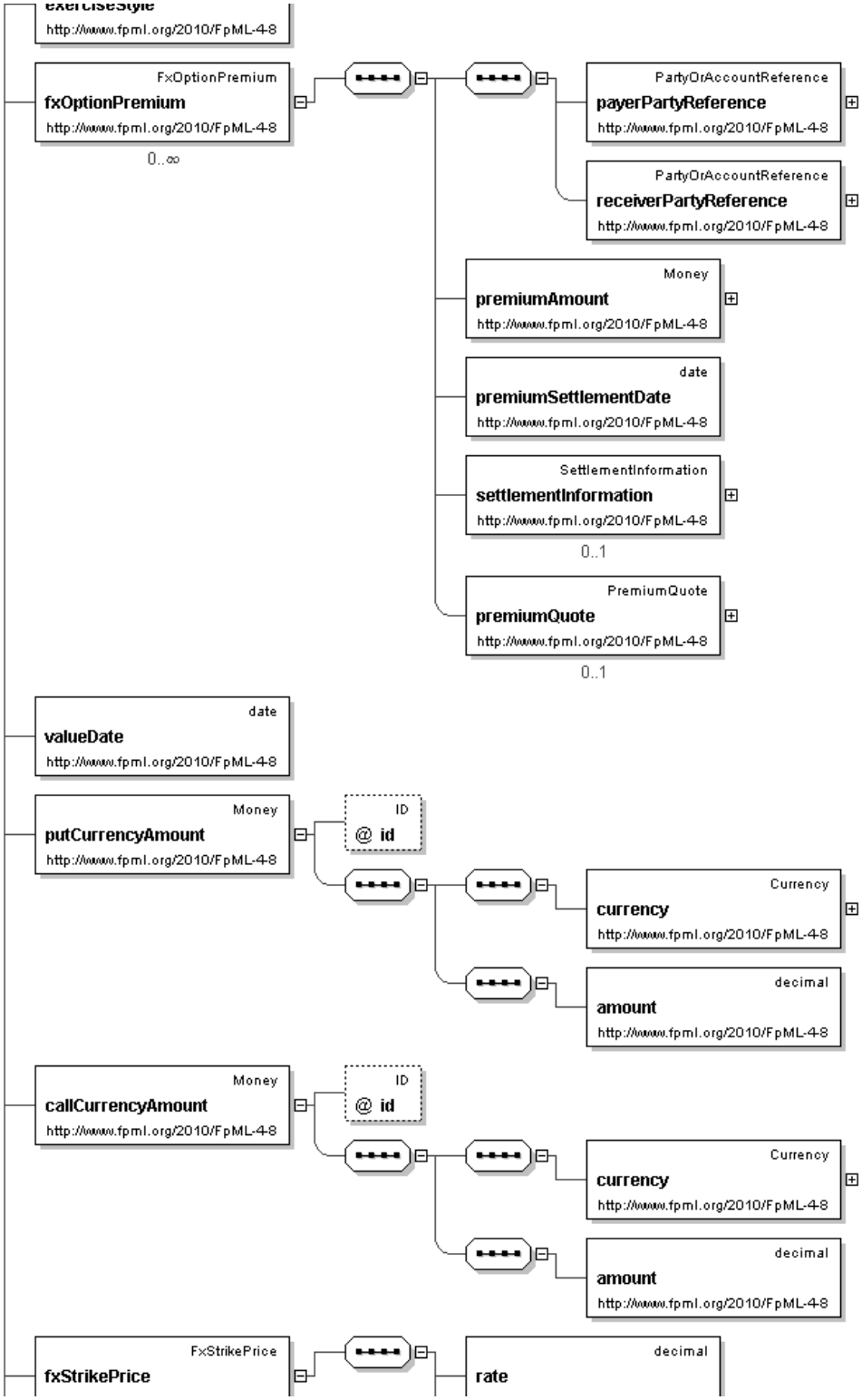
[Table of contents]

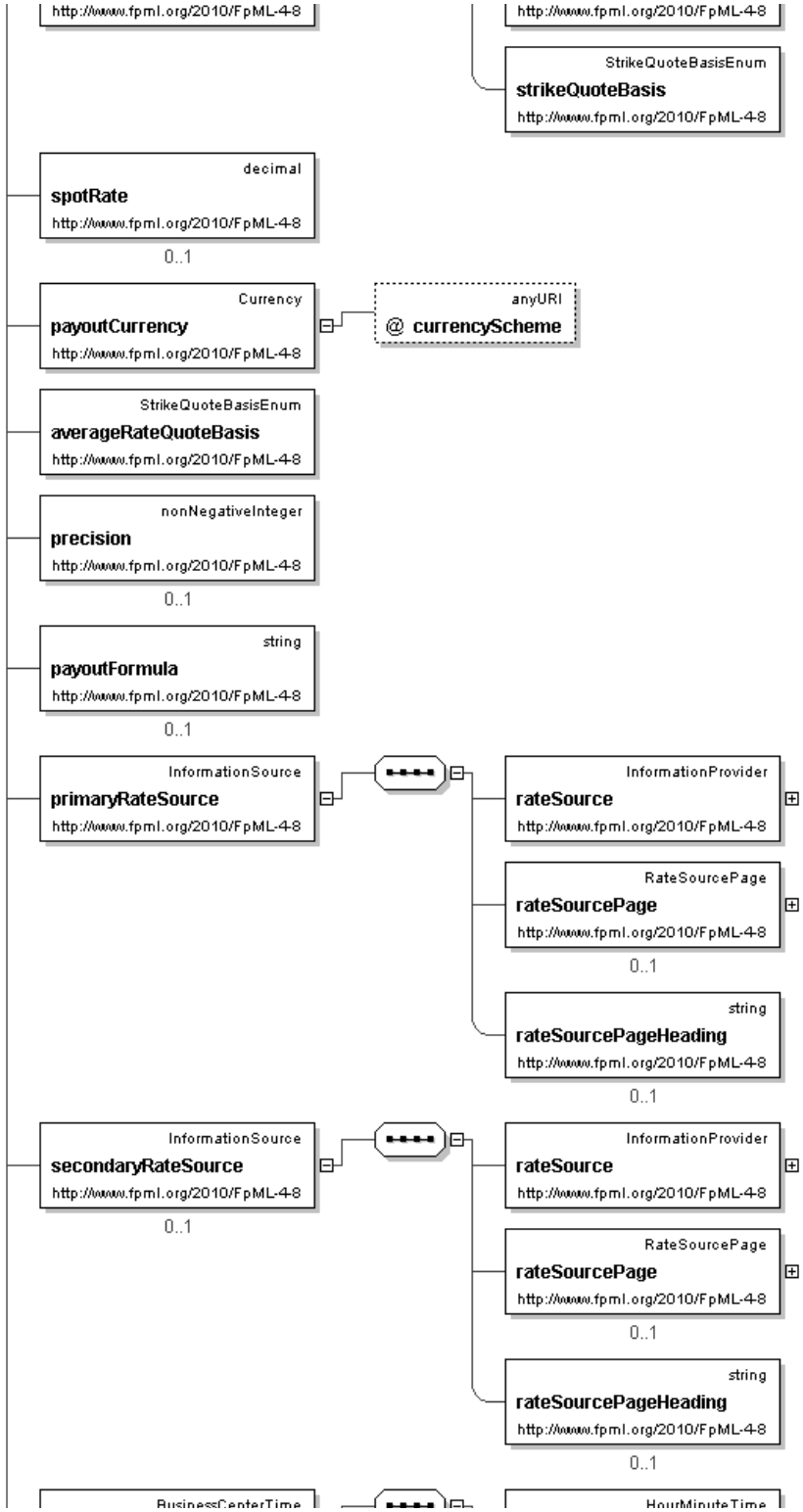
- *This element can be used wherever the following element is referenced:*
  - product

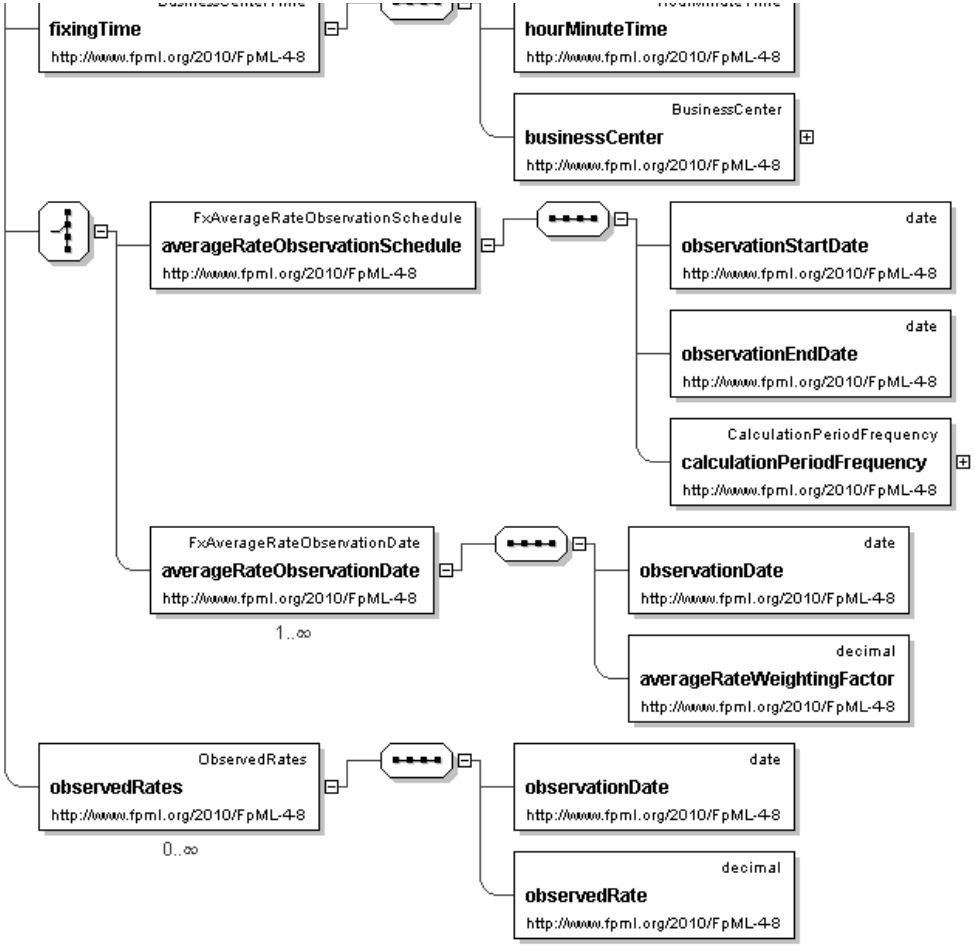
Name	fxAverageRateOption
Type	FxAverageRateOption
Nillable	no
Abstract	no
Documentation	A component describing an FX Average Rate Option product.

Logical Diagram




exerciseStyle
http://www.fpml.org/2010/FpML-4-8
FxOptionPremium
fxOptionPremium
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
premiumAmount
http://www.fpml.org/2010/FpML-4-8
date
premiumSettlementDate
http://www.fpml.org/2010/FpML-4-8
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumQuote
premiumQuote
http://www.fpml.org/2010/FpML-4-8
0..1
date
valueDate
http://www.fpml.org/2010/FpML-4-8
Money
putCurrencyAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
Money
callCurrencyAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
FxStrikePrice
fxStrikePrice
decimal
rate


http://www.fpml.org/2010/FpML-4-8
http://www.fpml.org/2010/FpML-4-8
StrikeQuoteBasisEnum
strikeQuoteBasis
http://www.fpml.org/2010/FpML-4-8
decimal
spotRate
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
payoutCurrency
http://www.fpml.org/2010/FpML-4-8
anyURI
@ currencyScheme
StrikeQuoteBasisEnum
averageRateQuoteBasis
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
precision
http://www.fpml.org/2010/FpML-4-8
0..1
string
payoutFormula
http://www.fpml.org/2010/FpML-4-8
0..1
InformationSource
primaryRateSource
http://www.fpml.org/2010/FpML-4-8
InformationProvider
rateSource
http://www.fpml.org/2010/FpML-4-8
RateSourcePage
rateSourcePage
http://www.fpml.org/2010/FpML-4-8
0..1
string
rateSourcePageHeading
http://www.fpml.org/2010/FpML-4-8
0..1
InformationSource
secondaryRateSource
http://www.fpml.org/2010/FpML-4-8
0..1
InformationProvider
rateSource
http://www.fpml.org/2010/FpML-4-8
RateSourcePage
rateSourcePage
http://www.fpml.org/2010/FpML-4-8
0..1
string
rateSourcePageHeading
http://www.fpml.org/2010/FpML-4-8
0..1

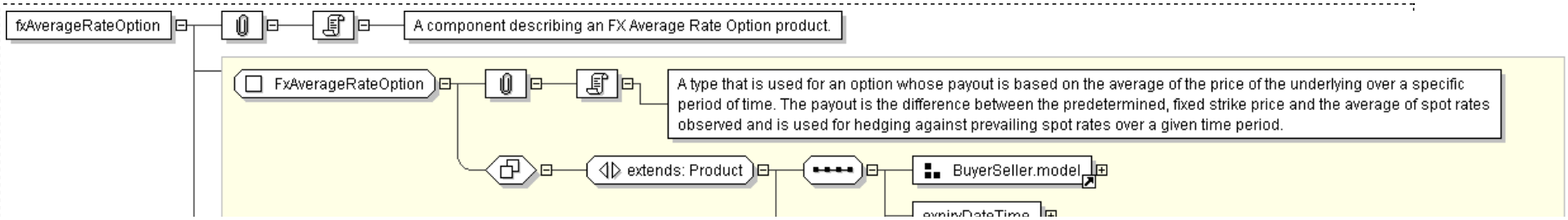


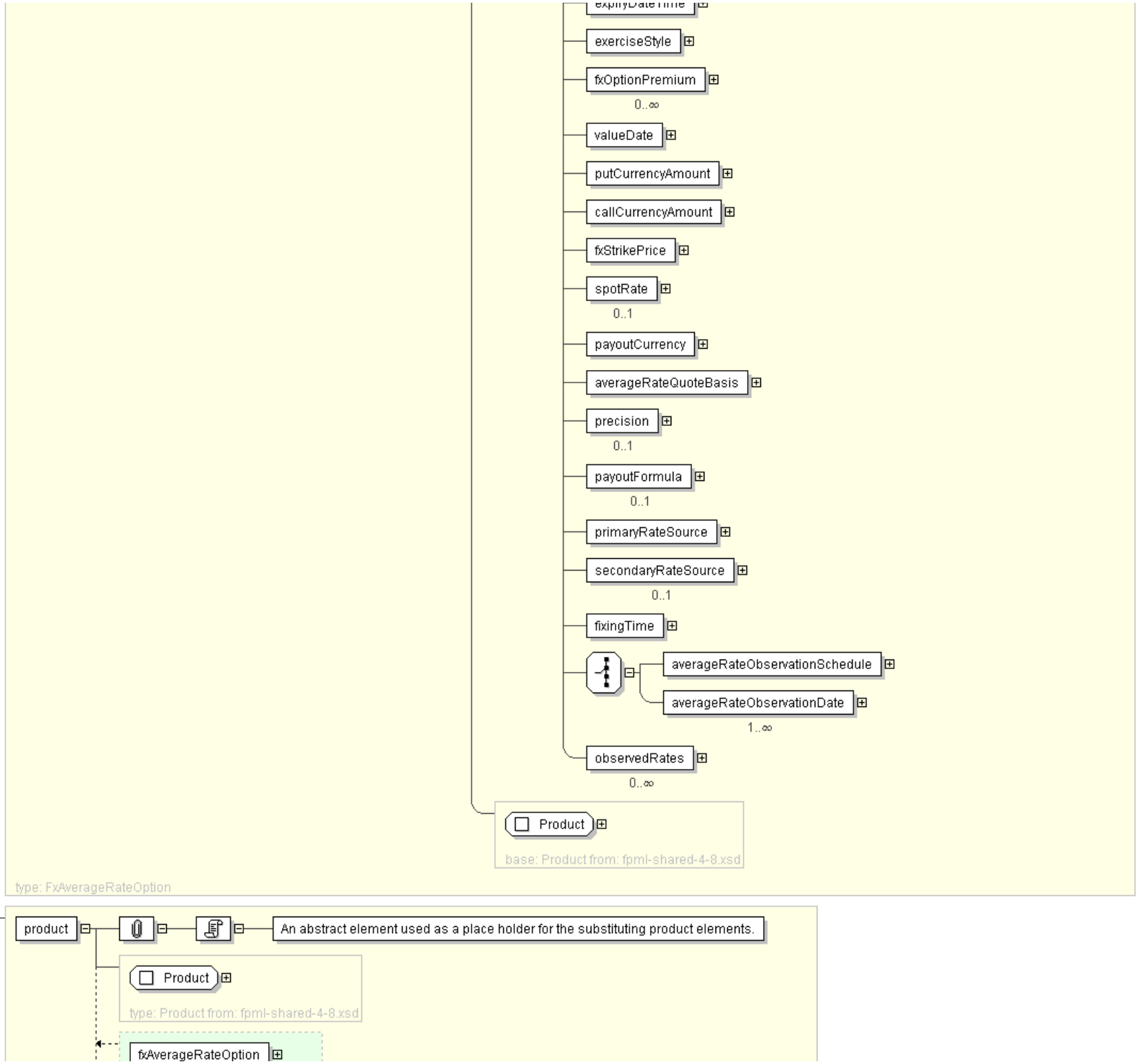
XML Instance Representation

```
<fxAverageRateOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that
    the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000
    ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in
    return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
    'The date and time in a location of the option expiry. In the case of american options this is the latest possible expiry date and
    time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
    'The manner in which the option can be exercised.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
```

```
    'Premium amount or premium installment amount for an option.'

    <valueDate> xsd:date </valueDate> [1]
    'The date on which both currencies traded will settle.'

    <putCurrencyAmount> Money </putCurrencyAmount> [1]
    'The currency amount that the option gives the right to sell.'

    <callCurrencyAmount> Money </callCurrencyAmount> [1]
    'The currency amount that the option gives the right to buy.'

    <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
    'TBA'

    <spotRate> xsd:decimal </spotRate> [0..1]
    'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX Forwards Market, this
    represents the current market rate for a particular currency pair. For barrier and digital/binary options, it can be useful to
    include the spot rate at the time the option was executed to make it easier to know whether the option needs to move \"up\" or
    \"down\" to be triggered.'

    <payoutCurrency> Currency </payoutCurrency> [1]
    'The ISO code of the currency in which a payout (if any) is to be made when a trigger is hit on a digital or barrier option.'

    <averageRateQuoteBasis> StrikeQuoteBasisEnum </averageRateQuoteBasis> [1]
    'The method by which the average rate that is being observed is quoted.'

    <precision> xsd:nonNegativeInteger </precision> [0..1]
    'Specifies the rounding precision in terms of a number of decimal places. Note how a percentage rate rounding of 5 decimal places is
    expressed as a rounding precision of 7 in the FpML document since the percentage is expressed as a decimal, e.g. 9.876543% (or
    0.09876543) being rounded to the nearest 5 decimal places is 9.87654% (or 0.0987654).'

    <payoutFormula> xsd:string </payoutFormula> [0..1]
    'The description of the mathematical computation for how the payout is computed.'

    <primaryRateSource> InformationSource </primaryRateSource> [1]
    'The primary source for where the rate observation will occur. Will typically be either a page or a reference bank published rate.'

    <secondaryRateSource> InformationSource </secondaryRateSource> [0..1]
    'An alternative, or secondary, source for where the rate observation will occur. Will typically be either a page or a reference bank
    published rate.'

    <fixingTime> BusinessCenterTime </fixingTime> [1]
    'The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g.
    11:00am London time.'

    Start Choice [1]
      <averageRateObservationSchedule> FxAverageRateObservationSchedule </averageRateObservationSchedule> [1]
      'Parametric schedule of rate observations.'

      <averageRateObservationDate> FxAverageRateObservationDate </averageRateObservationDate> [1..*]
      'One of more specific rate observation dates.'

    End Choice
    <observedRates> ObservedRates </observedRates> [0..*]
    'Describes prior rate observations within average rate options. Periodically, an average rate option agreement will be struck whereby
    some rates have already been observed in the past but will become part of computation of the average rate of the option. This
    structure provides for these previously observed rates to be included in the description of the trade.'

</fxAverageRateOption>
```

Diagram




expiryDateTime
exerciseStyle
fxOptionPremium
0..∞
valueDate
putCurrencyAmount
callCurrencyAmount
fxStrikePrice
spotRate
0..1
payoutCurrency
averageRateQuoteBasis
precision
0..1
payoutFormula
0..1
primaryRateSource
secondaryRateSource
0..1
fixingTime
averageRateObservationSchedule
averageRateObservationDate
1..∞
observedRates
0..∞
Product
base: Product from: fpml-shared-4-8.xsd
type: FxAverageRateOption
product
An abstract element used as a place holder for the substituting product elements.
Product
type: Product from: fpml-shared-4-8.xsd
fxAverageRateOption

Schema Component Representation

```
<xsd:element name="fxAverageRateOption" type=" FxAverageRateOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

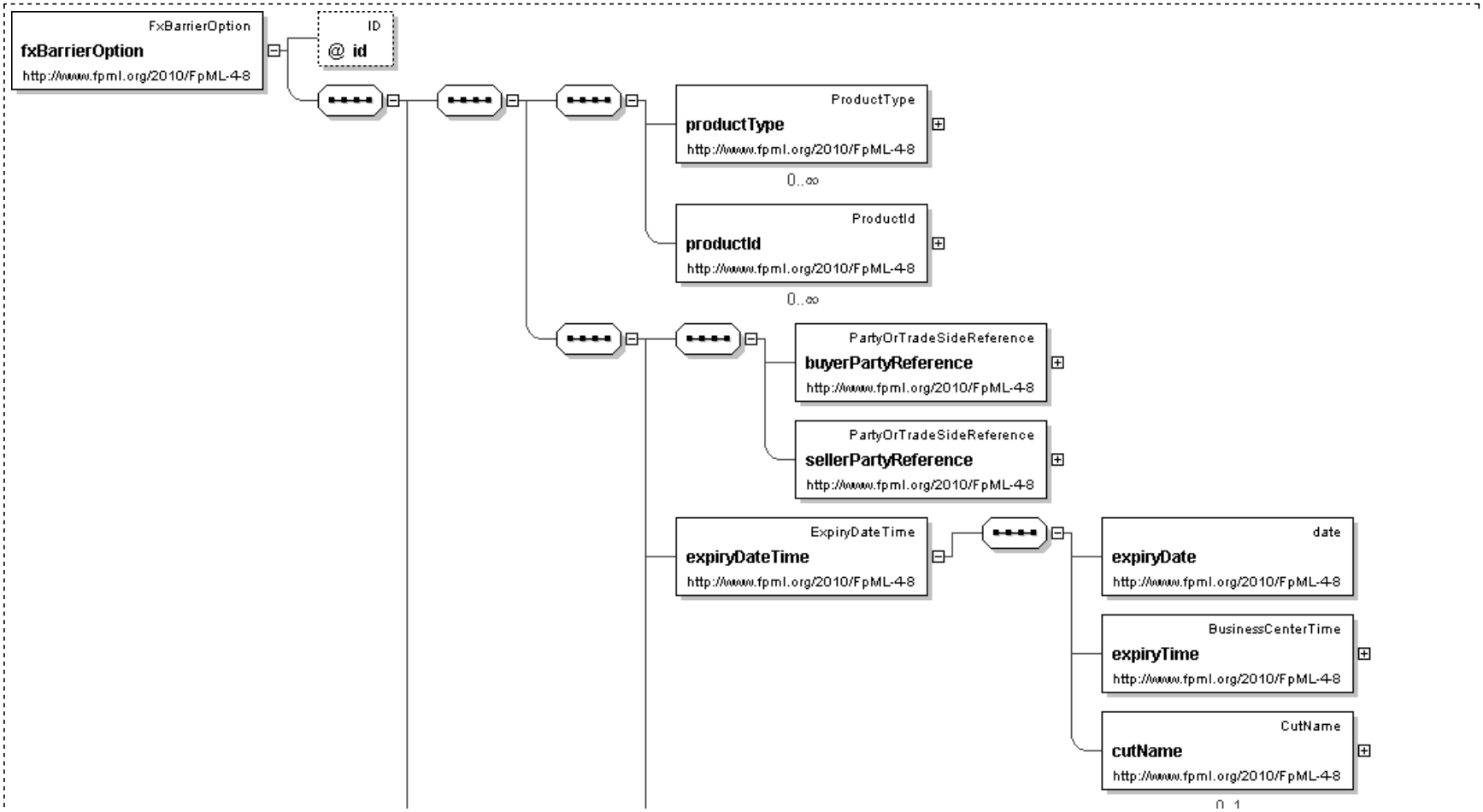
Element: fxBarrierOption

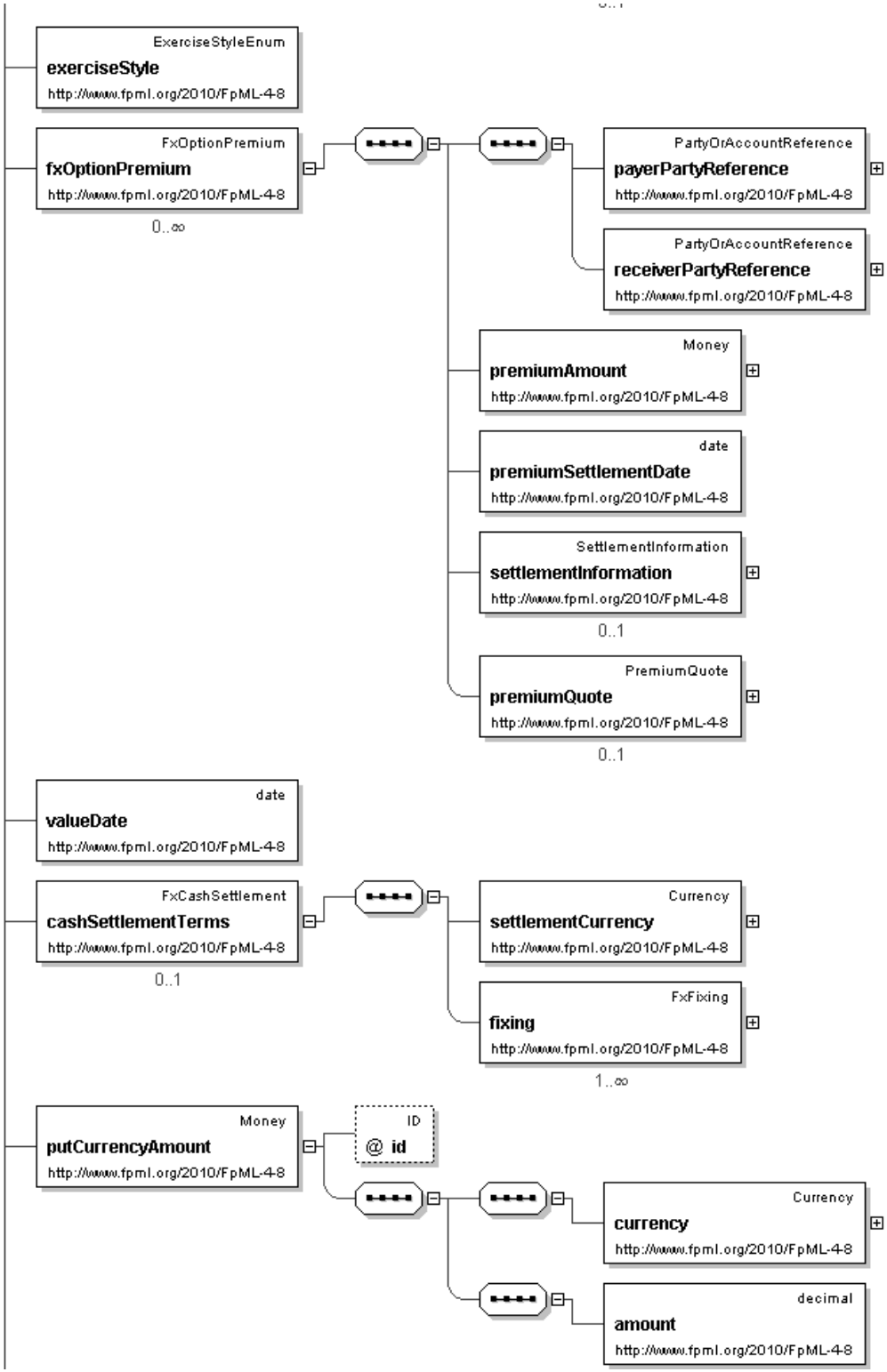
[Table of contents]

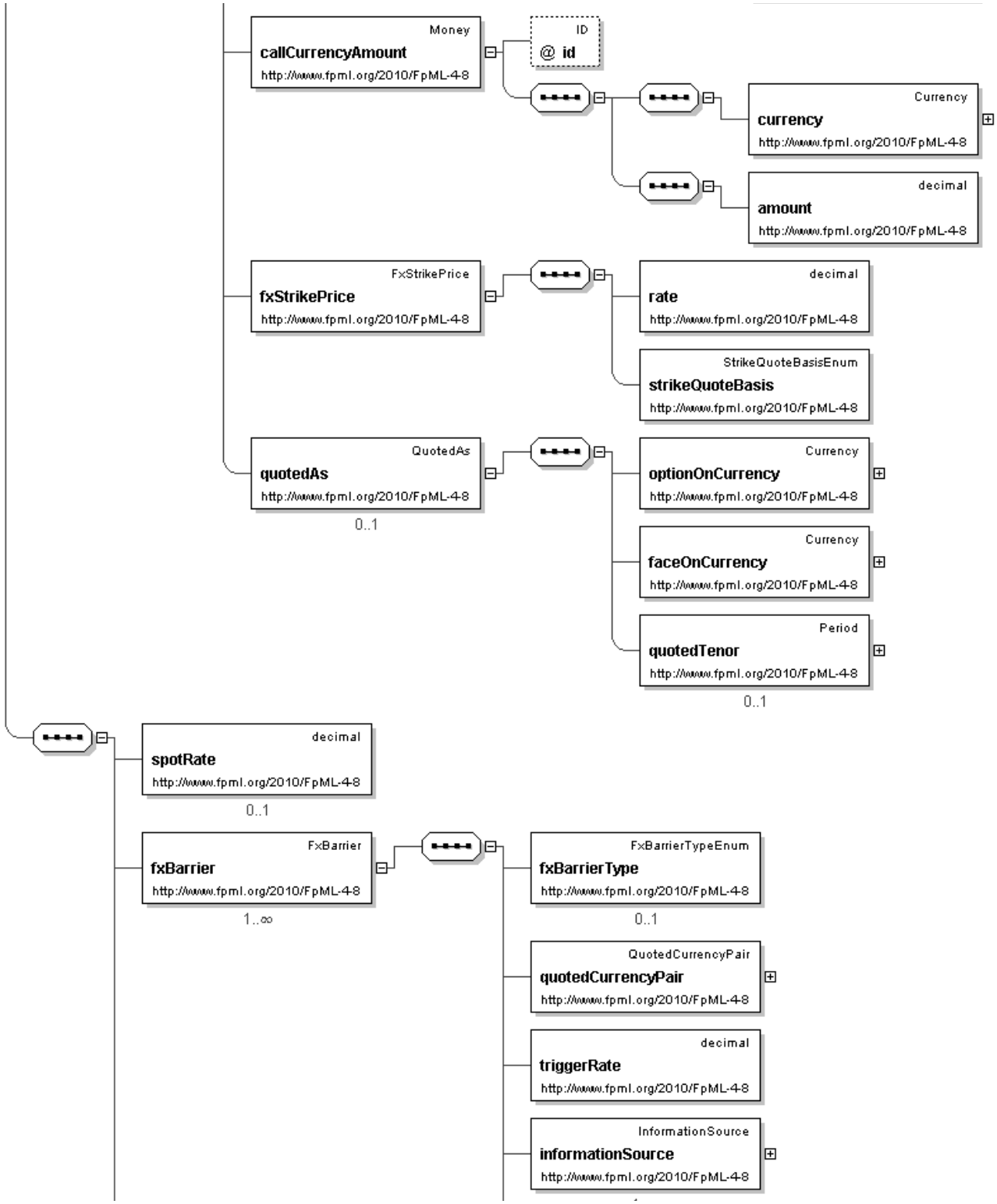
- *This element can be used wherever the following element is referenced:*
  - product

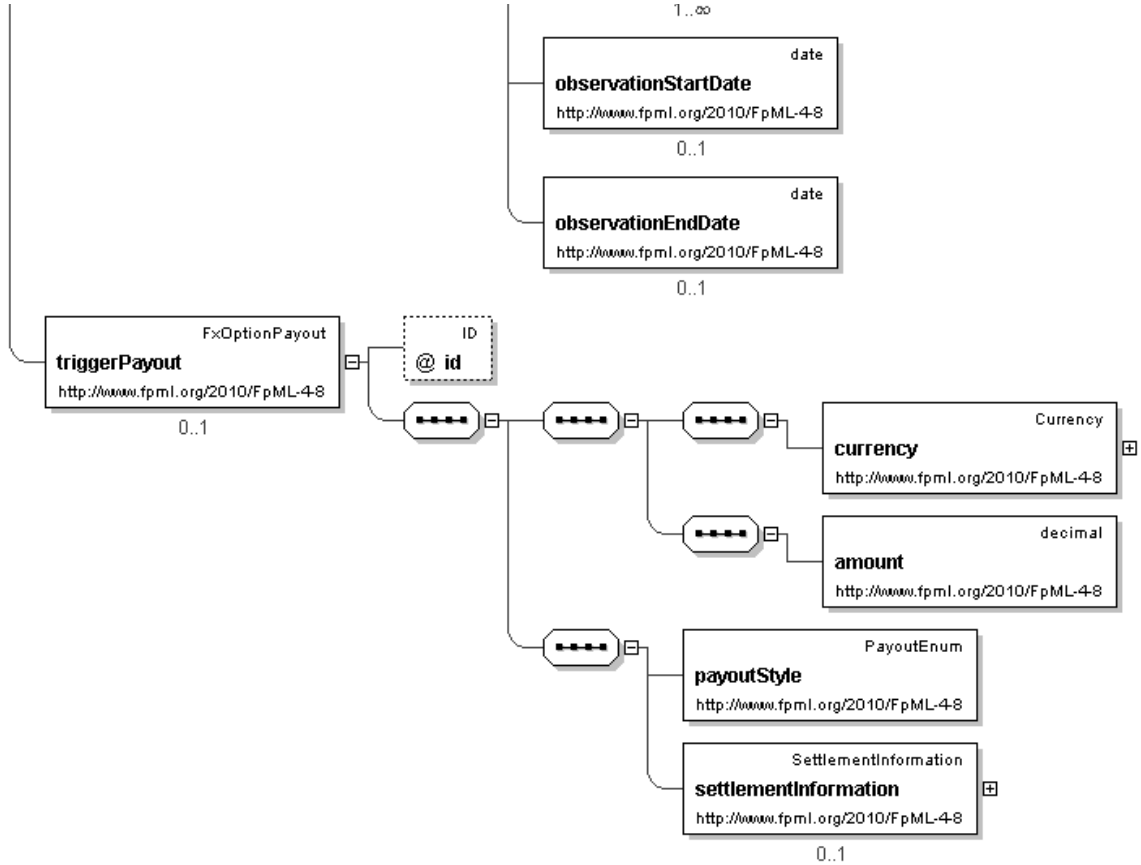
Name	fxBarrierOption
Type	FxBarrierOption
Nillable	no
Abstract	no
Documentation	A component describing a FX Barrier Option product.

Logical Diagram




ExerciseStyleEnum
exerciseStyle
http://www.fpml.org/2010/FpML-4-8
FxOptionPremium
fxOptionPremium
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
premiumAmount
http://www.fpml.org/2010/FpML-4-8
date
premiumSettlementDate
http://www.fpml.org/2010/FpML-4-8
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumQuote
premiumQuote
http://www.fpml.org/2010/FpML-4-8
0..1
date
valueDate
http://www.fpml.org/2010/FpML-4-8
FxCashSettlement
cashSettlementTerms
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
FxFixing
fixing
http://www.fpml.org/2010/FpML-4-8
1..∞
Money
putCurrencyAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8


Money
callCurrencyAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
FxStrikePrice
fxStrikePrice
http://www.fpml.org/2010/FpML-4-8
decimal
rate
http://www.fpml.org/2010/FpML-4-8
StrikeQuoteBasisEnum
strikeQuoteBasis
http://www.fpml.org/2010/FpML-4-8
QuotedAs
quotedAs
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
optionOnCurrency
http://www.fpml.org/2010/FpML-4-8
Currency
faceOnCurrency
http://www.fpml.org/2010/FpML-4-8
Period
quotedTenor
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
spotRate
http://www.fpml.org/2010/FpML-4-8
0..1
FxBarrier
fxBarrier
http://www.fpml.org/2010/FpML-4-8
1..∞
FxBarrierTypeEnum
fxBarrierType
http://www.fpml.org/2010/FpML-4-8
0..1
QuotedCurrencyPair
quotedCurrencyPair
http://www.fpml.org/2010/FpML-4-8
decimal
triggerRate
http://www.fpml.org/2010/FpML-4-8
InformationSource
informationSource
http://www.fpml.org/2010/FpML-4-8

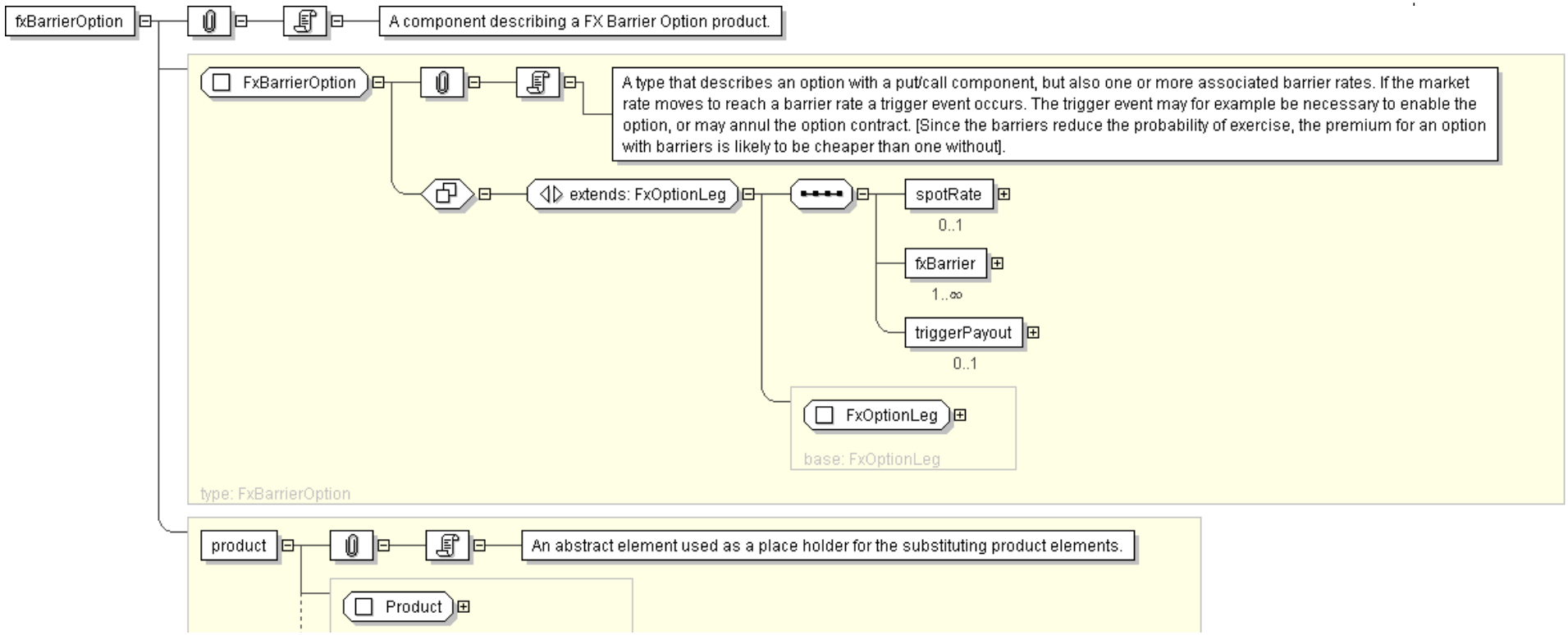


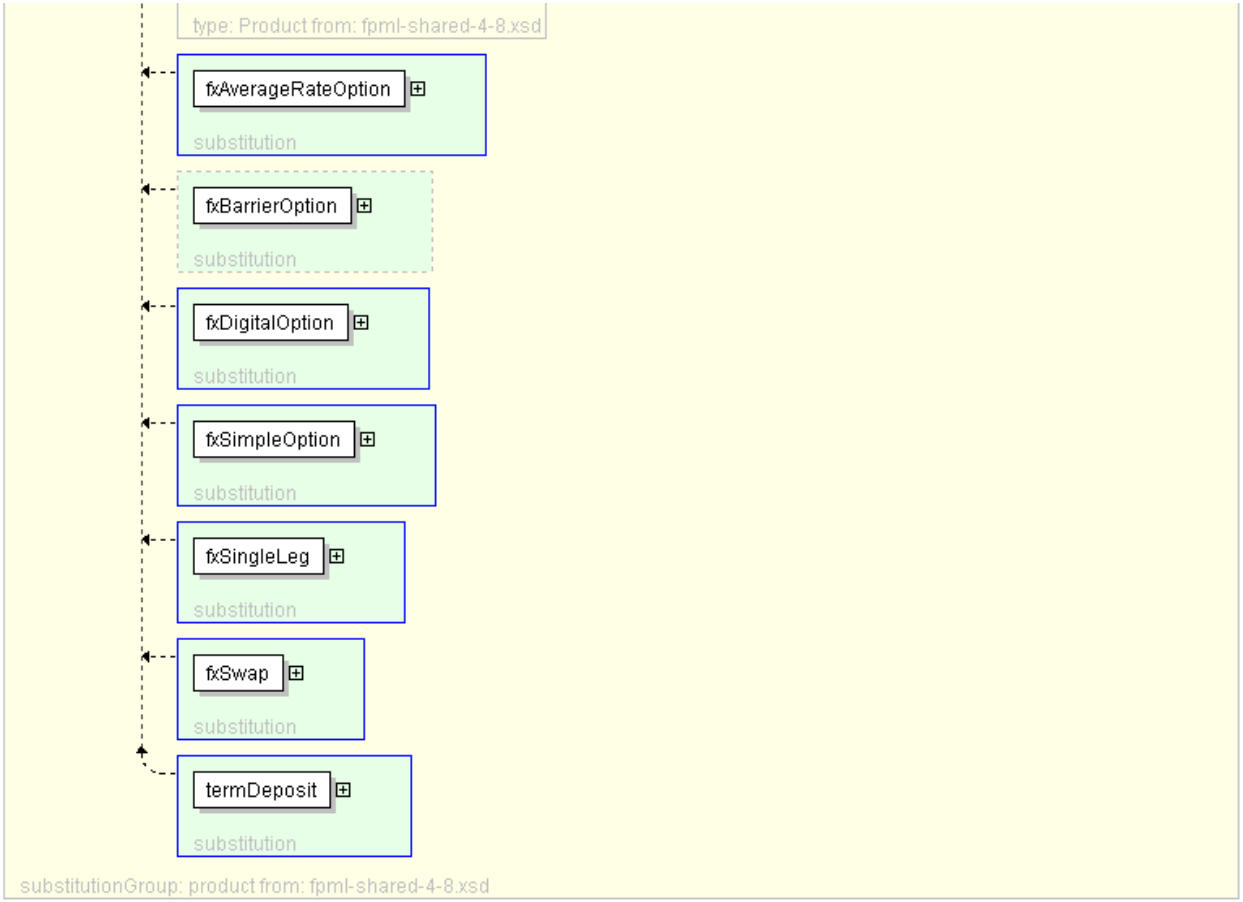
XML Instance Representation

```
<fxBarrierOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element.
   Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See
   2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and
   in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
   'The date and time in a location of the option expiry. In the case of american options this is the latest possible expiry date
   and time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
   'The manner in which the option can be exercised.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
   'Premium amount or premium installment amount for an option.'
```

```
    <valueDate> xsd:date </valueDate> [1]
    'The date on which both currencies traded will settle.'

    <cashSettlementTerms> FxCashSettlement </cashSettlementTerms> [0..1]
    'This optional element is only used if an option has been specified at execution time to be settled into a single cash
    payment. This would be used for a non-deliverable option.'

    <putCurrencyAmount> Money </putCurrencyAmount> [1]
    'The currency amount that the option gives the right to sell.'

    <callCurrencyAmount> Money </callCurrencyAmount> [1]
    'The currency amount that the option gives the right to buy.'

    <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
    'TBA'

    <quotedAs> QuotedAs </quotedAs> [0..1]
    'Describes how the option was quoted.'

    <spotRate> xsd:decimal </spotRate> [0..1]
    'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX Forwards Market,
    this represents the current market rate for a particular currency pair. For barrier and digital/binary options, it can be
    useful to include the spot rate at the time the option was executed to make it easier to know whether the option needs to move
    \"up\" or \"down\" to be triggered.'

    <fxBarrier> FxBarrier </fxBarrier> [1..*]
    'Information about a barrier rate in a Barrier Option - specifying the exact criteria for a trigger event to occur.'

    <triggerPayout> FxOptionPayout </triggerPayout> [0..1]
    'The amount of currency which becomes payable if and when a trigger event occurs.'

</fxBarrierOption>
```

Diagram





Schema Component Representation

```
<xsd:element name="fxBarrierOption" type=" FxBarrierOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

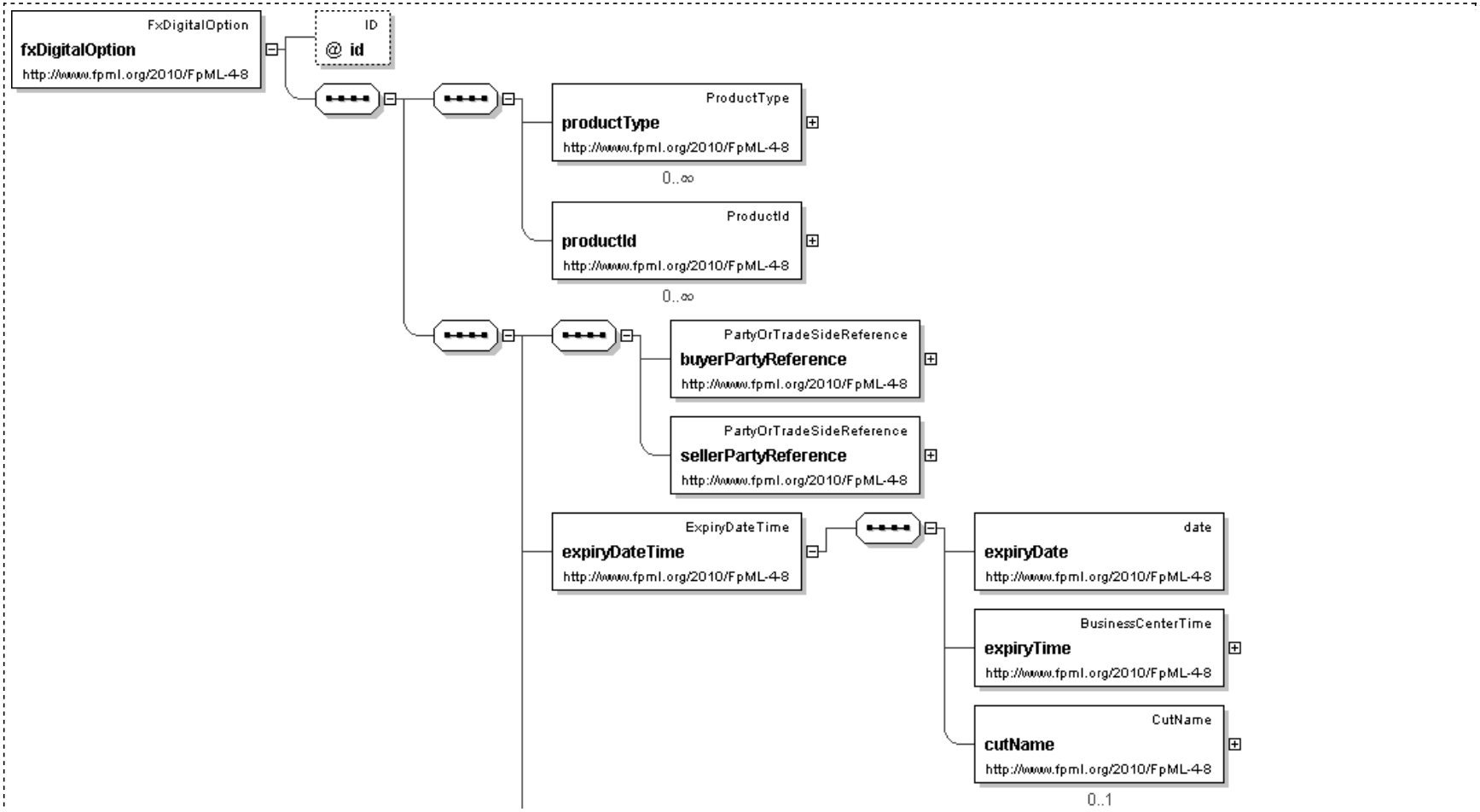
Element: fxDigitalOption

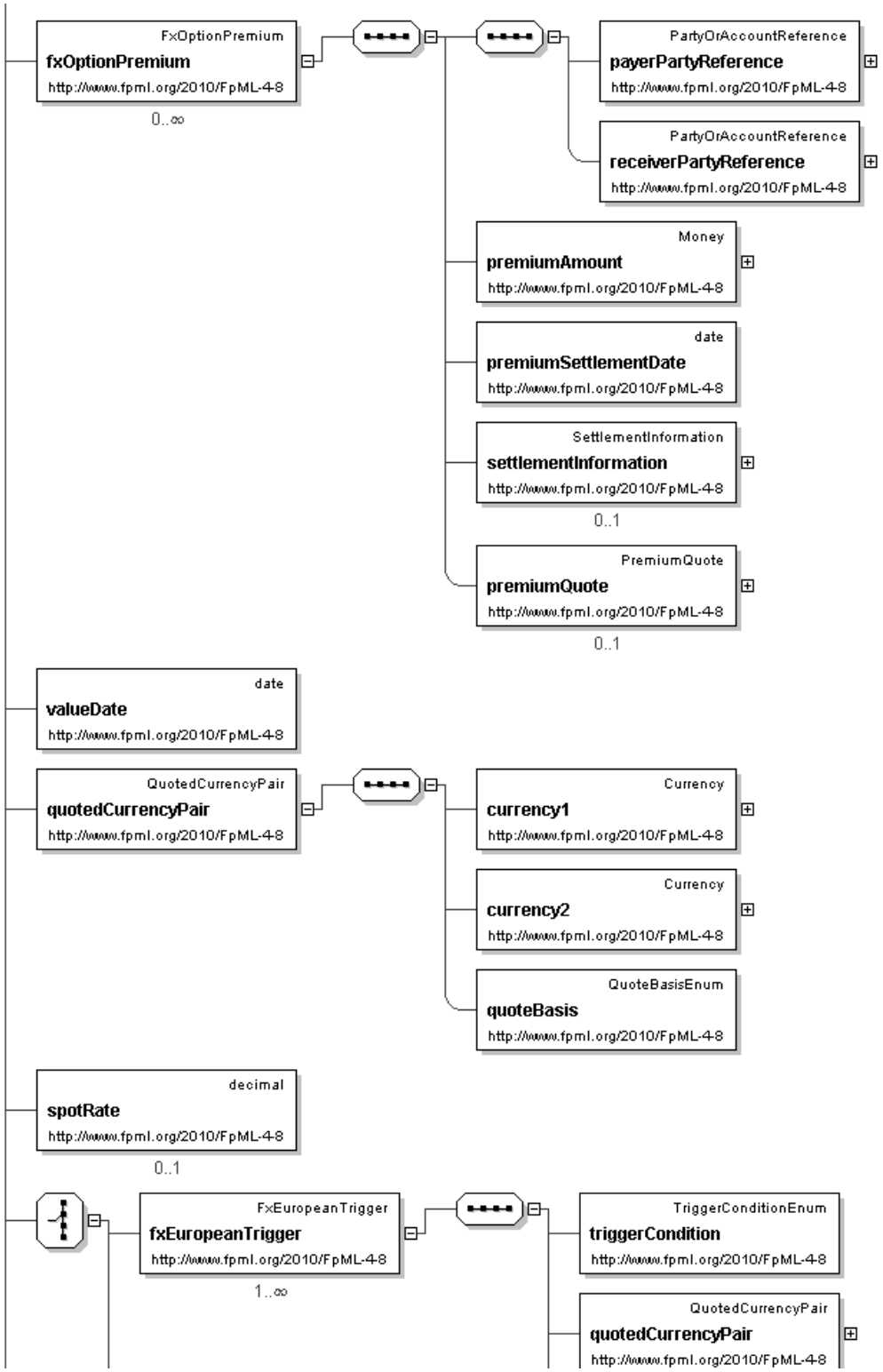
[Table of contents]

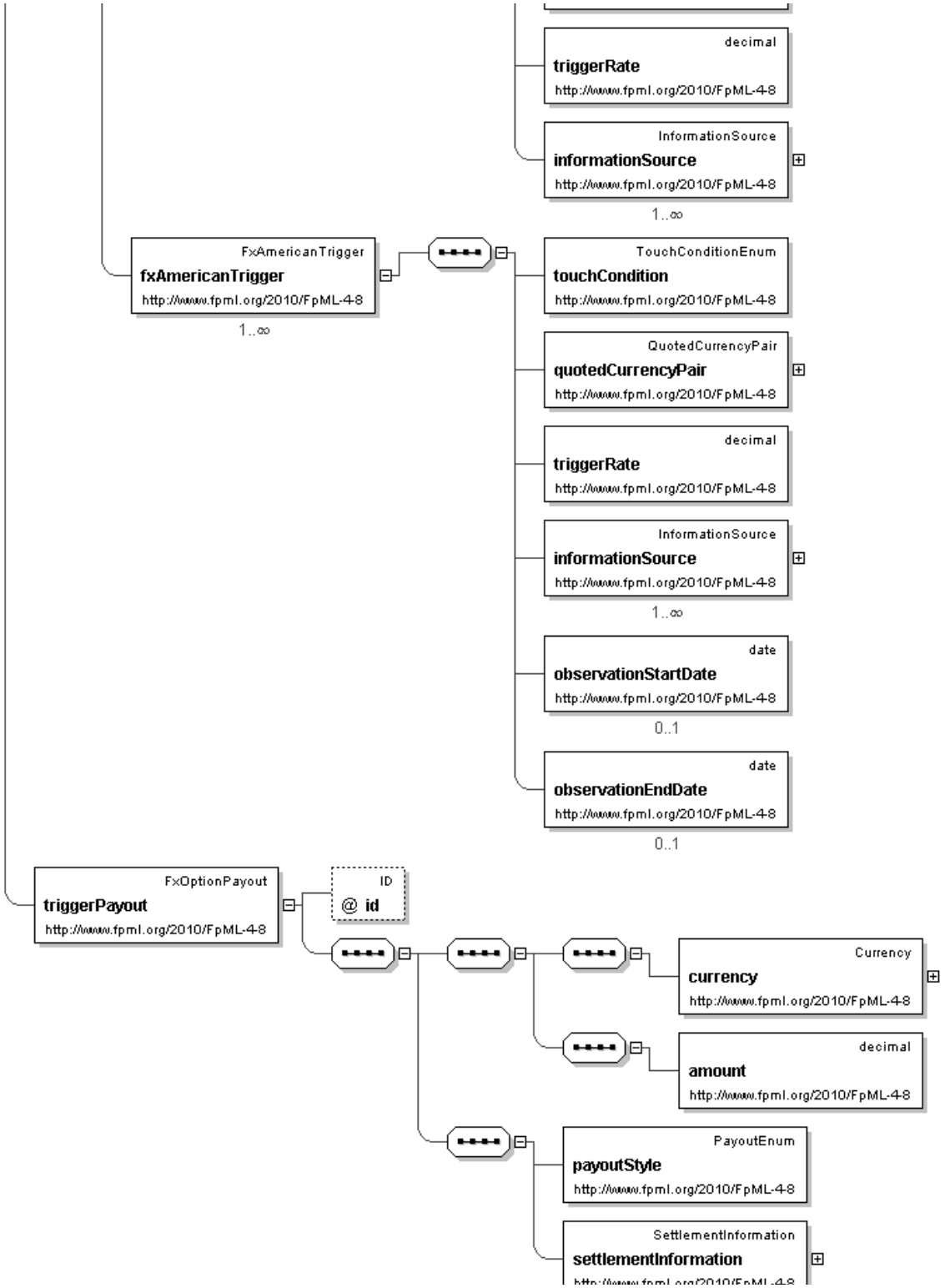
- *This element can be used wherever the following element is referenced:*
  - product

Name	fxDigitalOption
Type	FxDigitalOption
Nillable	no
Abstract	no
Documentation	A component describing a FX Digital Option product.

Logical Diagram




FxOptionPremium
fxOptionPremium
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
premiumAmount
http://www.fpml.org/2010/FpML-4-8
date
premiumSettlementDate
http://www.fpml.org/2010/FpML-4-8
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumQuote
premiumQuote
http://www.fpml.org/2010/FpML-4-8
0..1
date
valueDate
http://www.fpml.org/2010/FpML-4-8
QuotedCurrencyPair
quotedCurrencyPair
http://www.fpml.org/2010/FpML-4-8
Currency
currency1
http://www.fpml.org/2010/FpML-4-8
Currency
currency2
http://www.fpml.org/2010/FpML-4-8
QuoteBasisEnum
quoteBasis
http://www.fpml.org/2010/FpML-4-8
decimal
spotRate
http://www.fpml.org/2010/FpML-4-8
0..1
FxEuropeanTrigger
fxEuropeanTrigger
http://www.fpml.org/2010/FpML-4-8
1..∞
TriggerConditionEnum
triggerCondition
http://www.fpml.org/2010/FpML-4-8
QuotedCurrencyPair
quotedCurrencyPair
http://www.fpml.org/2010/FpML-4-8

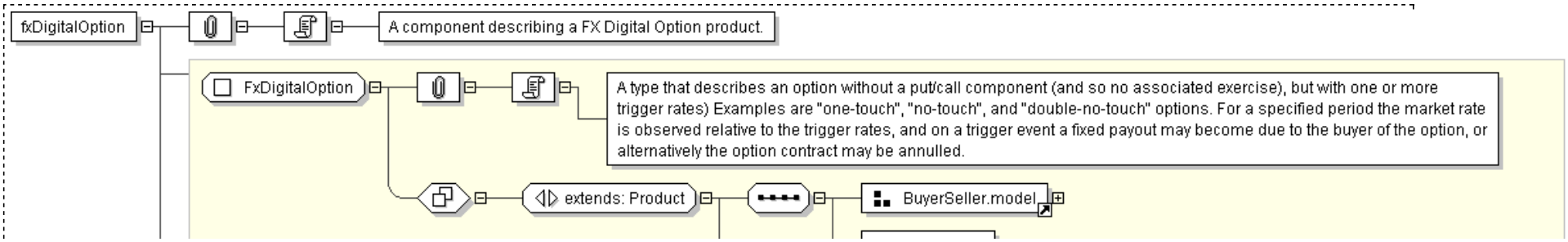

decimal
triggerRate
http://www.fpml.org/2010/FpML-4-8
InformationSource
informationSource
http://www.fpml.org/2010/FpML-4-8
1..∞
FxAmericanTrigger
fxAmericanTrigger
http://www.fpml.org/2010/FpML-4-8
1..∞
TouchConditionEnum
touchCondition
http://www.fpml.org/2010/FpML-4-8
QuotedCurrencyPair
quotedCurrencyPair
http://www.fpml.org/2010/FpML-4-8
decimal
triggerRate
http://www.fpml.org/2010/FpML-4-8
InformationSource
informationSource
http://www.fpml.org/2010/FpML-4-8
1..∞
date
observationStartDate
http://www.fpml.org/2010/FpML-4-8
0..1
date
observationEndDate
http://www.fpml.org/2010/FpML-4-8
0..1
FxOptionPayout
triggerPayout
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8
PayoutEnum
payoutStyle
http://www.fpml.org/2010/FpML-4-8
SettlementInformation
settlementInformation

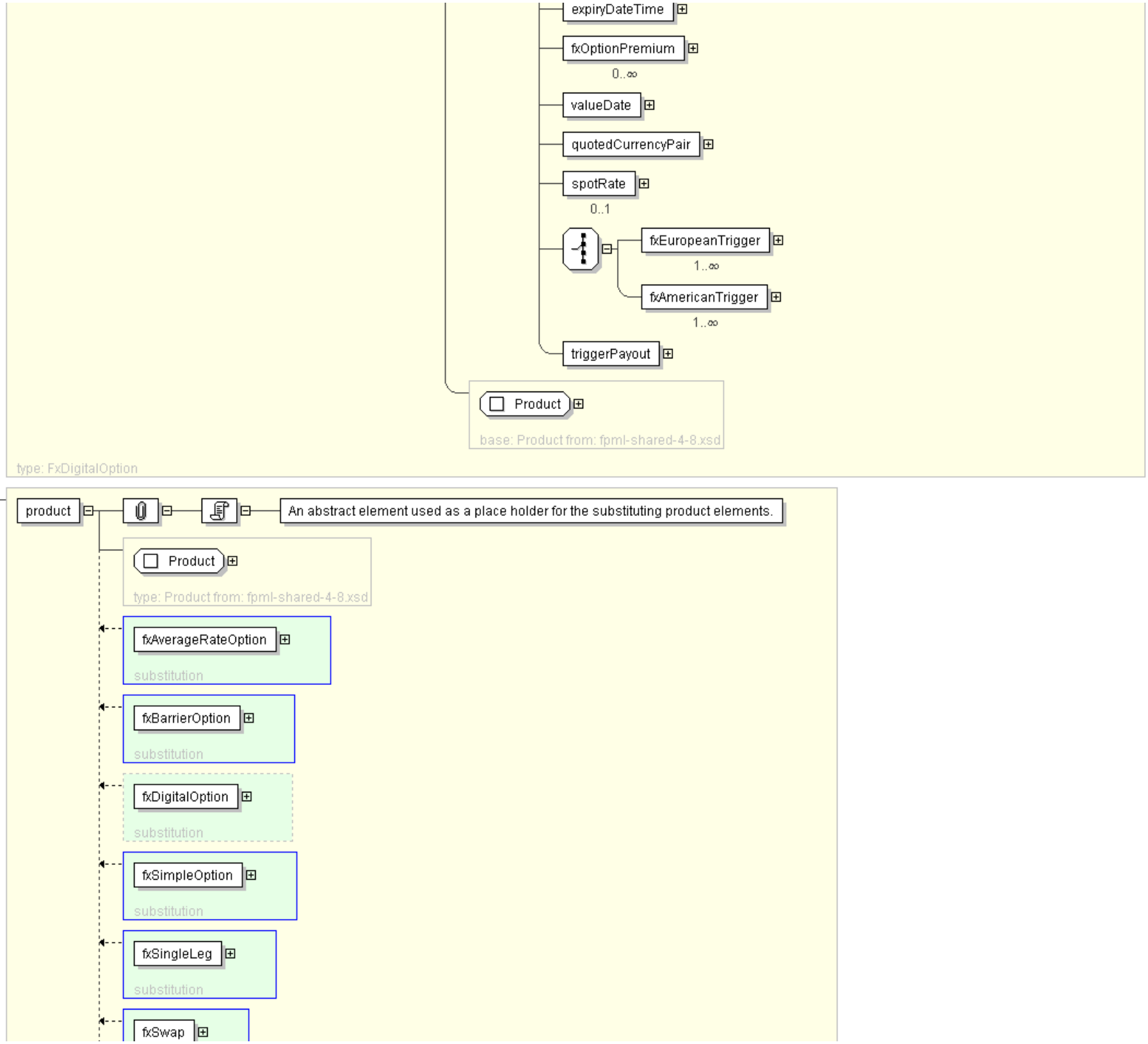
0..1

XML Instance Representation

```
<fxDigitalOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
    that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See
    2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and
    in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
    payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
    'The date and time in a location of the option expiry. In the case of american options this is the latest possible expiry date
    and time.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
    'Premium amount or premium installment amount for an option.'

   <valueDate> xsd:date </valueDate> [1]
    'The date on which both currencies traded will settle.'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
    'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <spotRate> xsd:decimal </spotRate> [0..1]
    'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX Forwards Market,
    this represents the current market rate for a particular currency pair. For barrier and digital/binary options, it can be
    useful to include the spot rate at the time the option was executed to make it easier to know whether the option needs to move
    \"up\" or \"down\" to be triggered.'

   Start Choice [1]
      <fxEuropeanTrigger> FxEuropeanTrigger </fxEuropeanTrigger> [1..*]
       'A European trigger occurs if the trigger criteria are met, but these are valid (and an observation is made) only at the
       maturity of the option.'

      <fxAmericanTrigger> FxAmericanTrigger </fxAmericanTrigger> [1..*]
       'An American trigger occurs if the trigger criteria are met at any time from the initiation to the maturity of the option.'

   End Choice
   <triggerPayout> FxOptionPayout </triggerPayout> [1]
    'The amount of currency which becomes payable if and when a trigger event occurs.'

</fxDigitalOption>
```

Diagram




expiryDateTime
fxOptionPremium
0..∞
valueDate
quotedCurrencyPair
spotRate
0..1
fxEuropeanTrigger
1..∞
fxAmericanTrigger
1..∞
triggerPayout
Product
base: Product from: fpml-shared-4-8.xsd
type: FxDigitalOption
product
An abstract element used as a place holder for the substituting product elements.
Product
type: Product from: fpml-shared-4-8.xsd
fxAverageRateOption
substitution
fxBarrierOption
substitution
fxDigitalOption
substitution
fxSimpleOption
substitution
fxSingleLeg
substitution
fxSwap

Schema Component Representation

```
<xsd:element name="fxDigitalOption" type=" FxDigitalOption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

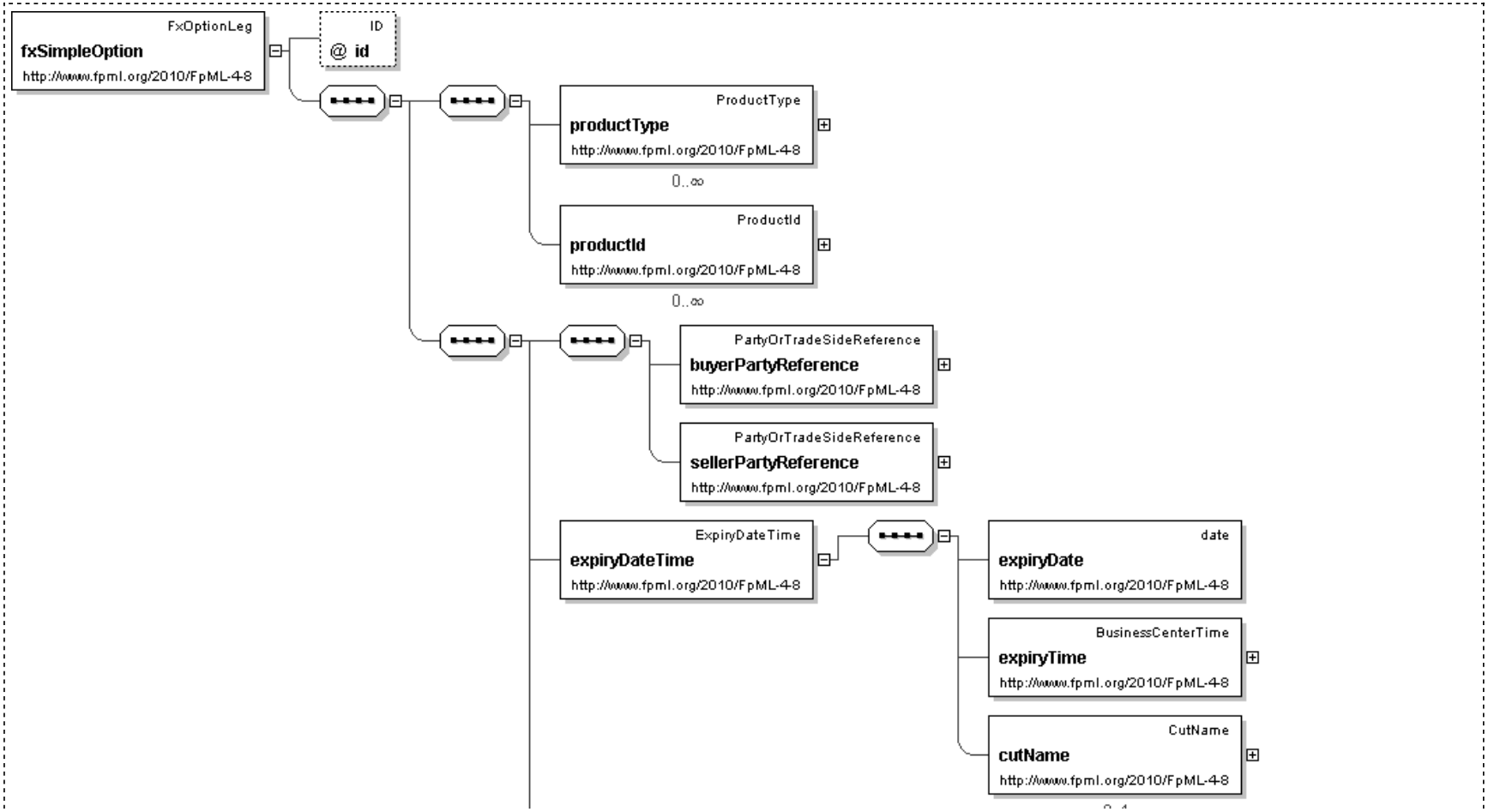
Element: fxSimpleOption

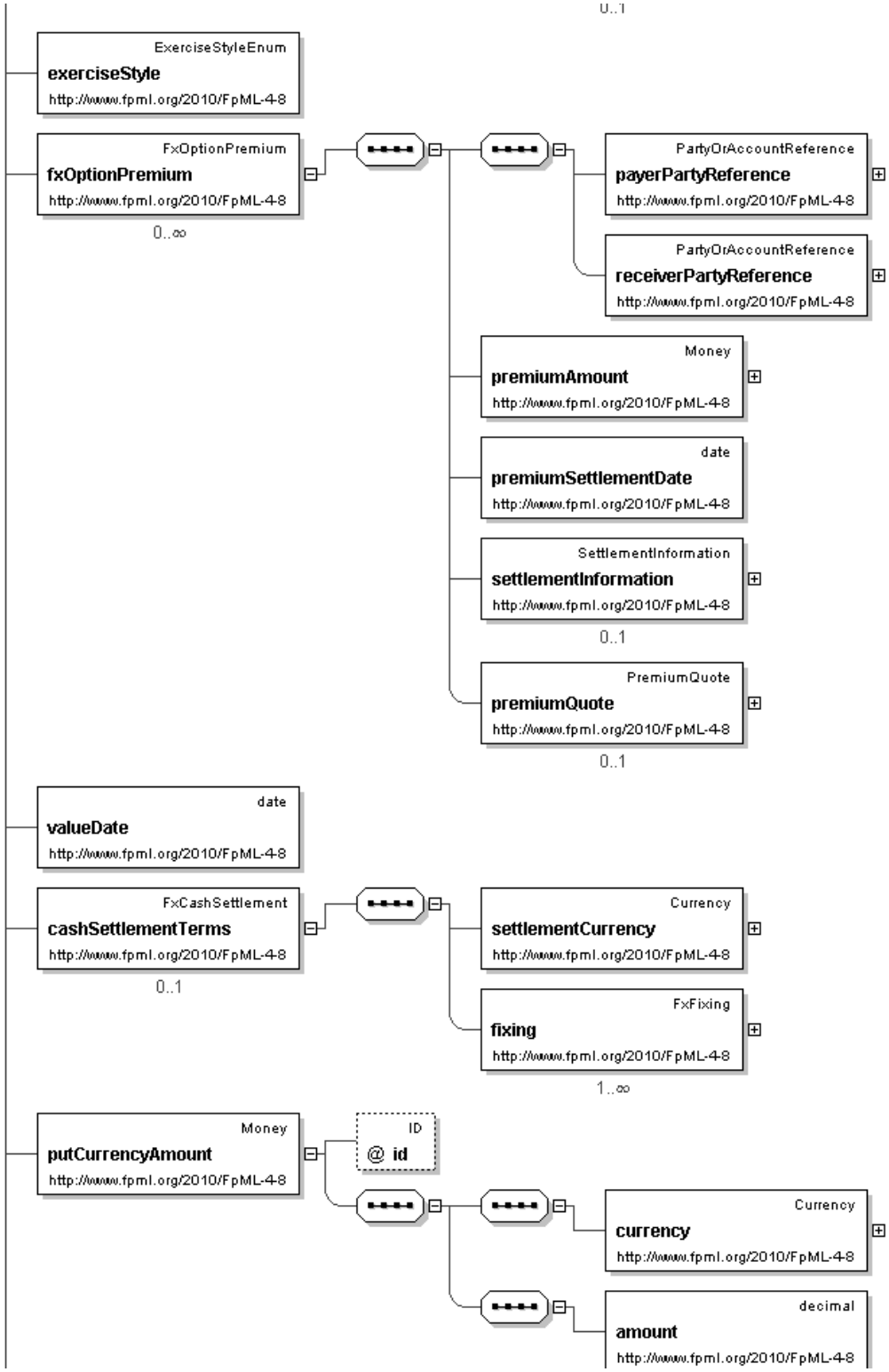
[Table of contents]

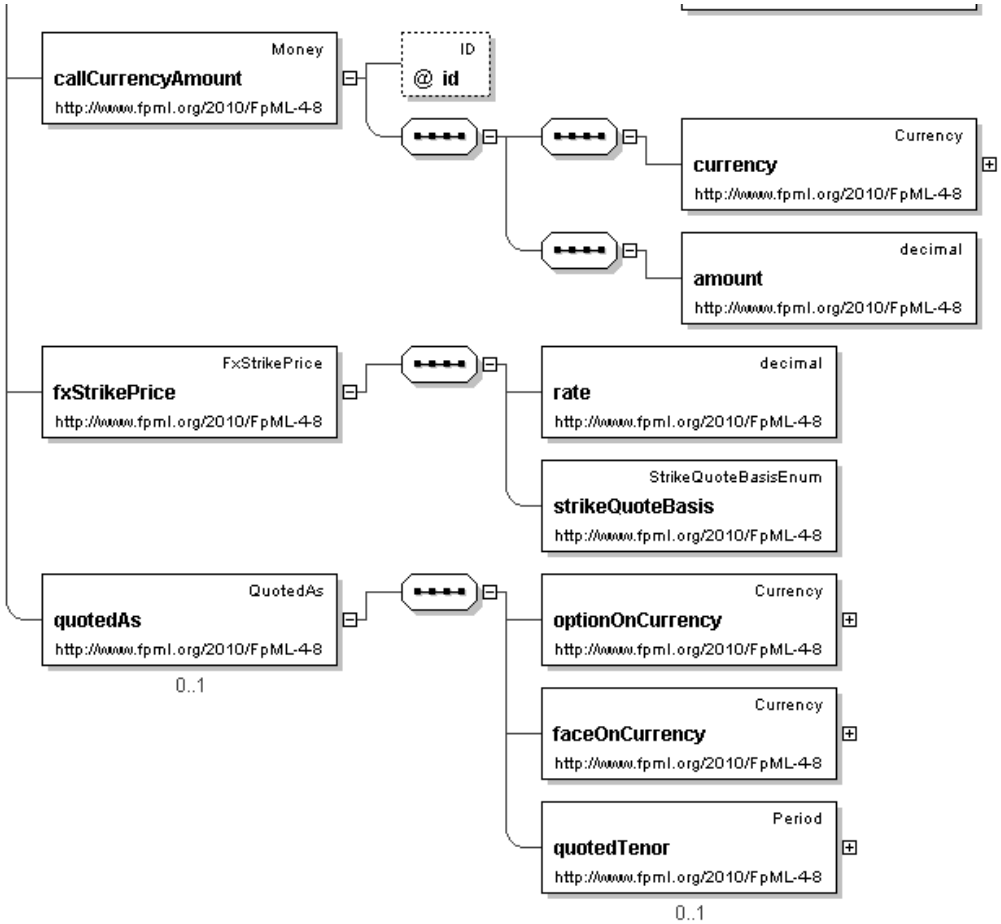
- *This element can be used wherever the following element is referenced:*
  - product

Name	fxSimpleOption
Type	FxOptionLeg
Nillable	no
Abstract	no
Documentation	A component describing a FX Simple Option product

Logical Diagram




0..1
ExerciseStyleEnum
exerciseStyle
http://www.fpml.org/2010/FpML-4-8
FxOptionPremium
fxOptionPremium
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
premiumAmount
http://www.fpml.org/2010/FpML-4-8
date
premiumSettlementDate
http://www.fpml.org/2010/FpML-4-8
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
PremiumQuote
premiumQuote
http://www.fpml.org/2010/FpML-4-8
0..1
date
valueDate
http://www.fpml.org/2010/FpML-4-8
FxCashSettlement
cashSettlementTerms
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
settlementCurrency
http://www.fpml.org/2010/FpML-4-8
FxFixing
fixing
http://www.fpml.org/2010/FpML-4-8
1..∞
Money
putCurrencyAmount
http://www.fpml.org/2010/FpML-4-8
ID
@ id
Currency
currency
http://www.fpml.org/2010/FpML-4-8
decimal
amount
http://www.fpml.org/2010/FpML-4-8

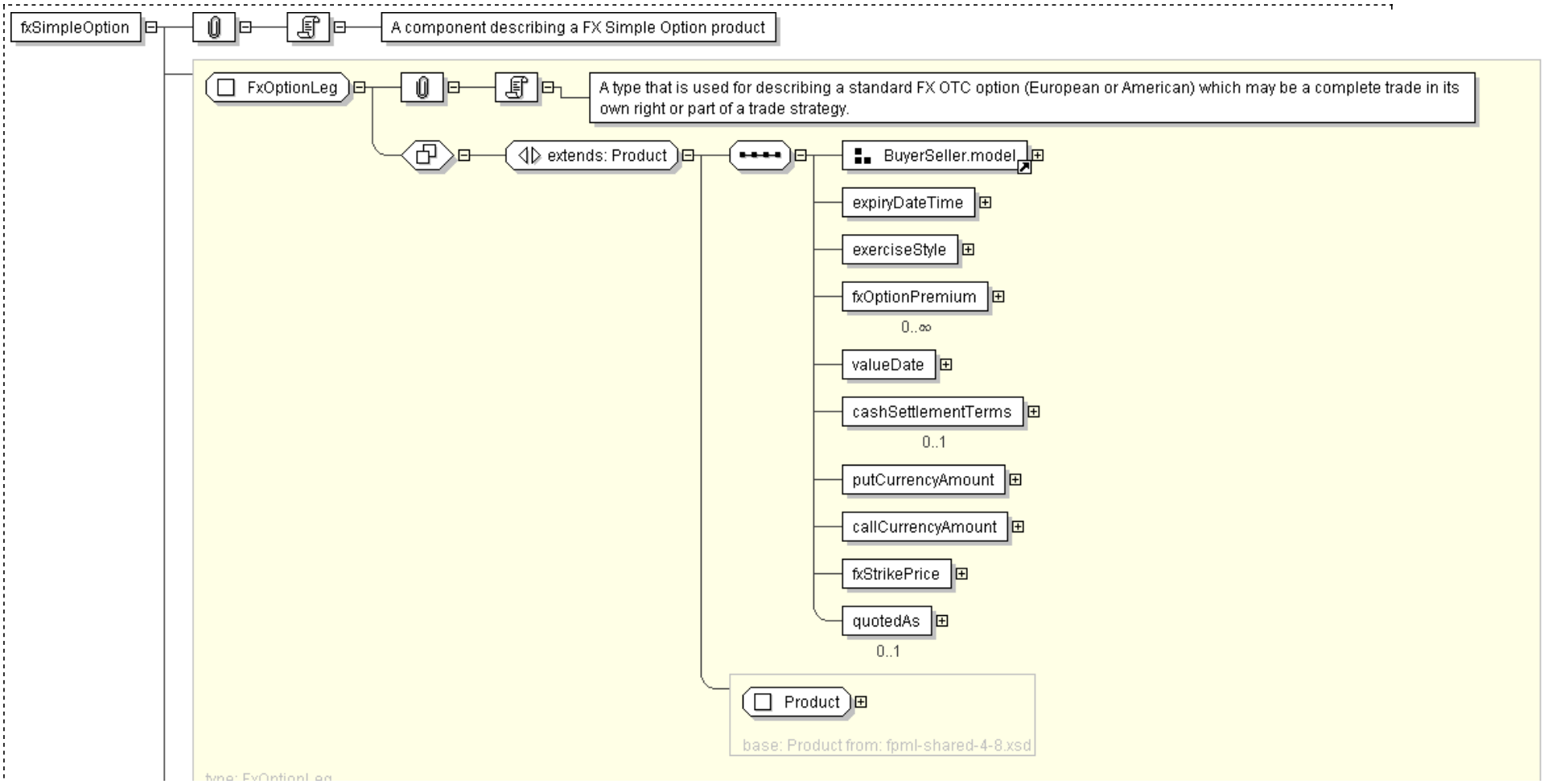


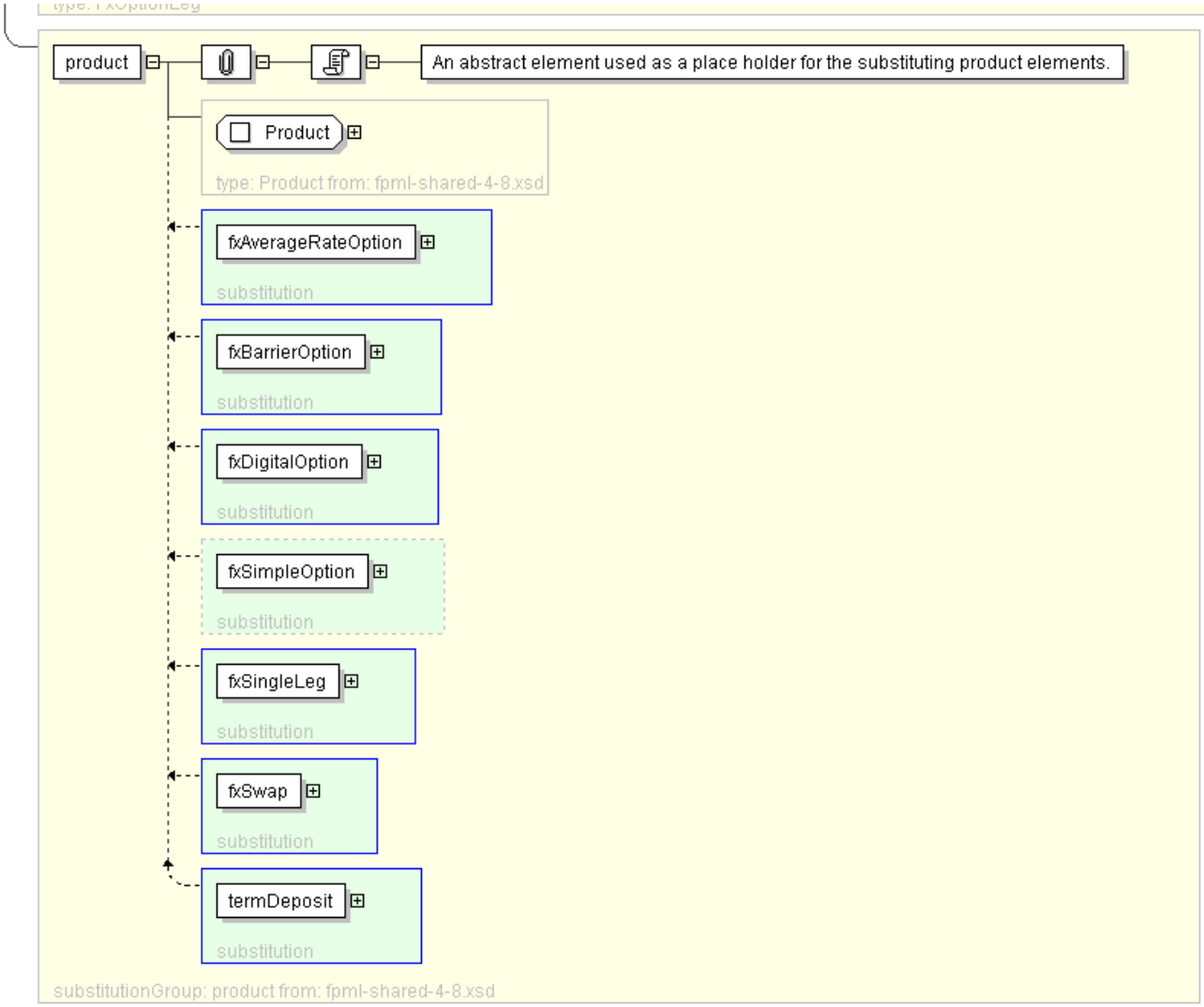
XML Instance Representation

```
<fxSimpleOption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element.
    Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See
    2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and
    in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating
    rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
    'The date and time in a location of the option expiry. In the case of american options this is the latest possible expiry
    date and time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
    'The manner in which the option can be exercised.'
```

```
    <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
    'Premium amount or premium installment amount for an option.'

    <valueDate> xsd:date </valueDate> [1]
    'The date on which both currencies traded will settle.'

    <cashSettlementTerms> FxCashSettlement </cashSettlementTerms> [0..1]
    'This optional element is only used if an option has been specified at execution time to be settled into a single cash
    payment. This would be used for a non-deliverable option.'

    <putCurrencyAmount> Money </putCurrencyAmount> [1]
    'The currency amount that the option gives the right to sell.'

    <callCurrencyAmount> Money </callCurrencyAmount> [1]
    'The currency amount that the option gives the right to buy.'

    <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
    'TBA'

    <quotedAs> QuotedAs </quotedAs> [0..1]
    'Describes how the option was quoted.'

</fxSimpleOption>
```

Diagram





Schema Component Representation

```
<xsd:element name="fxSimpleOption" type=" FxOptionLeg " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

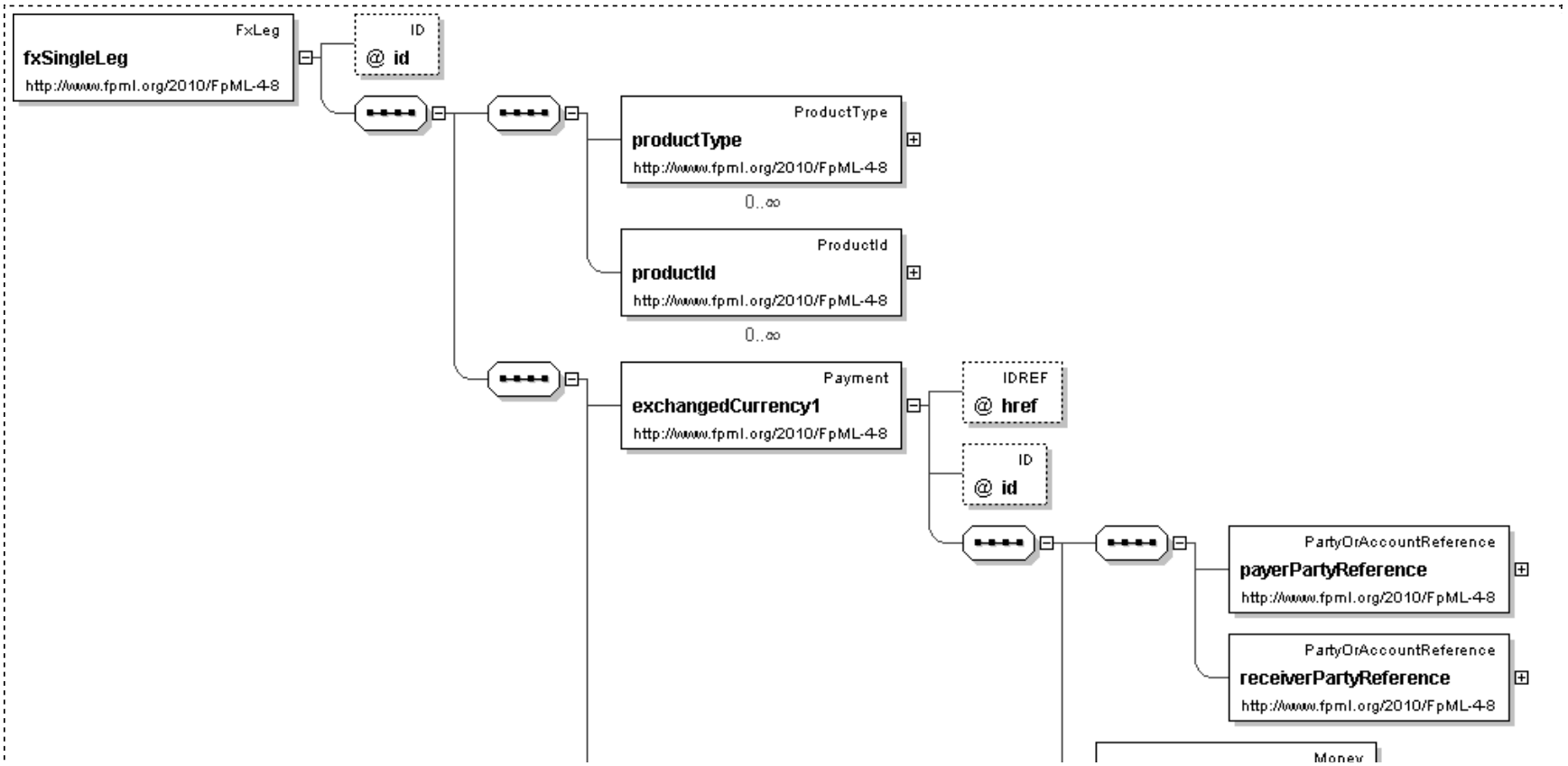
Element: fxSingleLeg

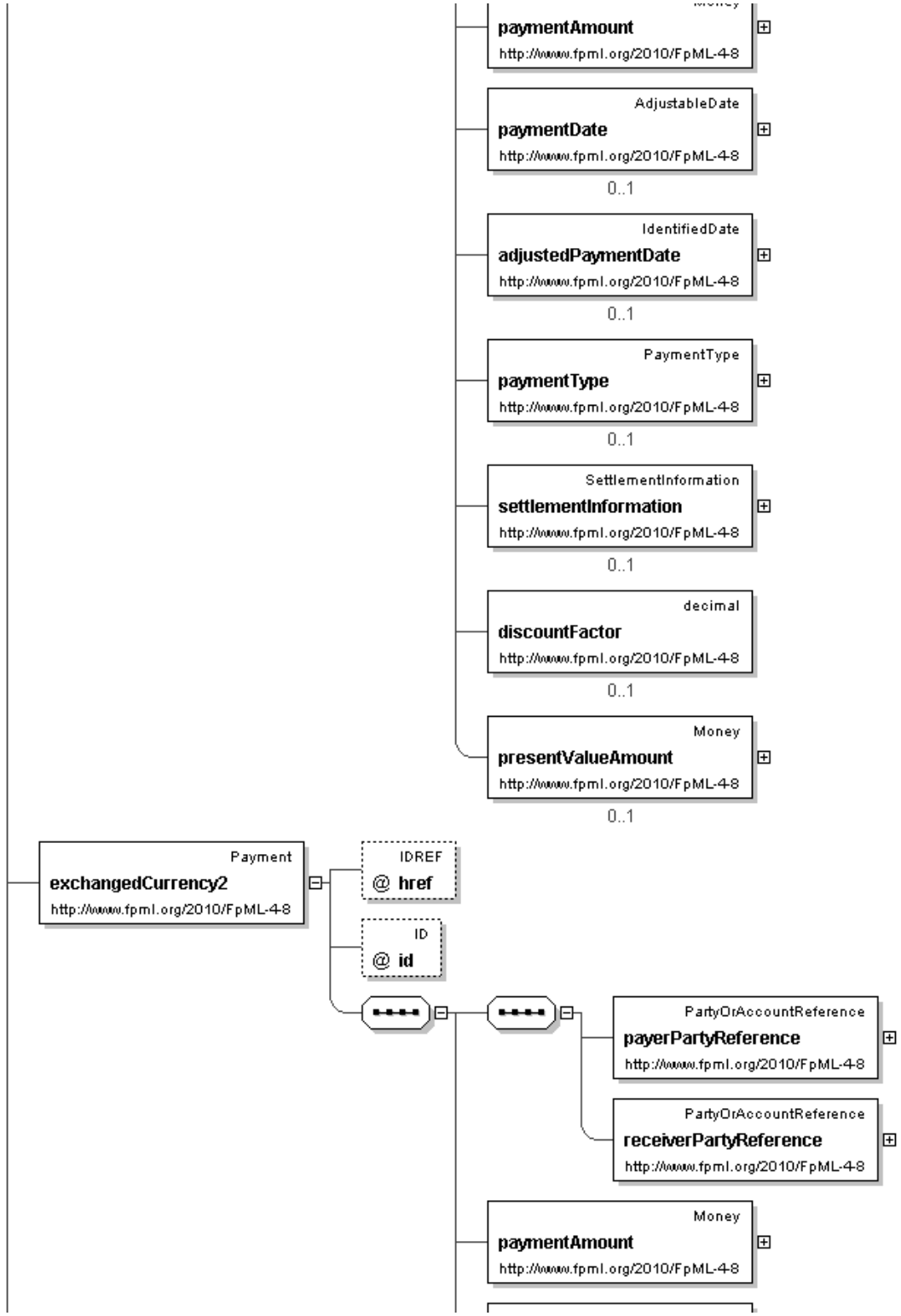
[Table of contents]

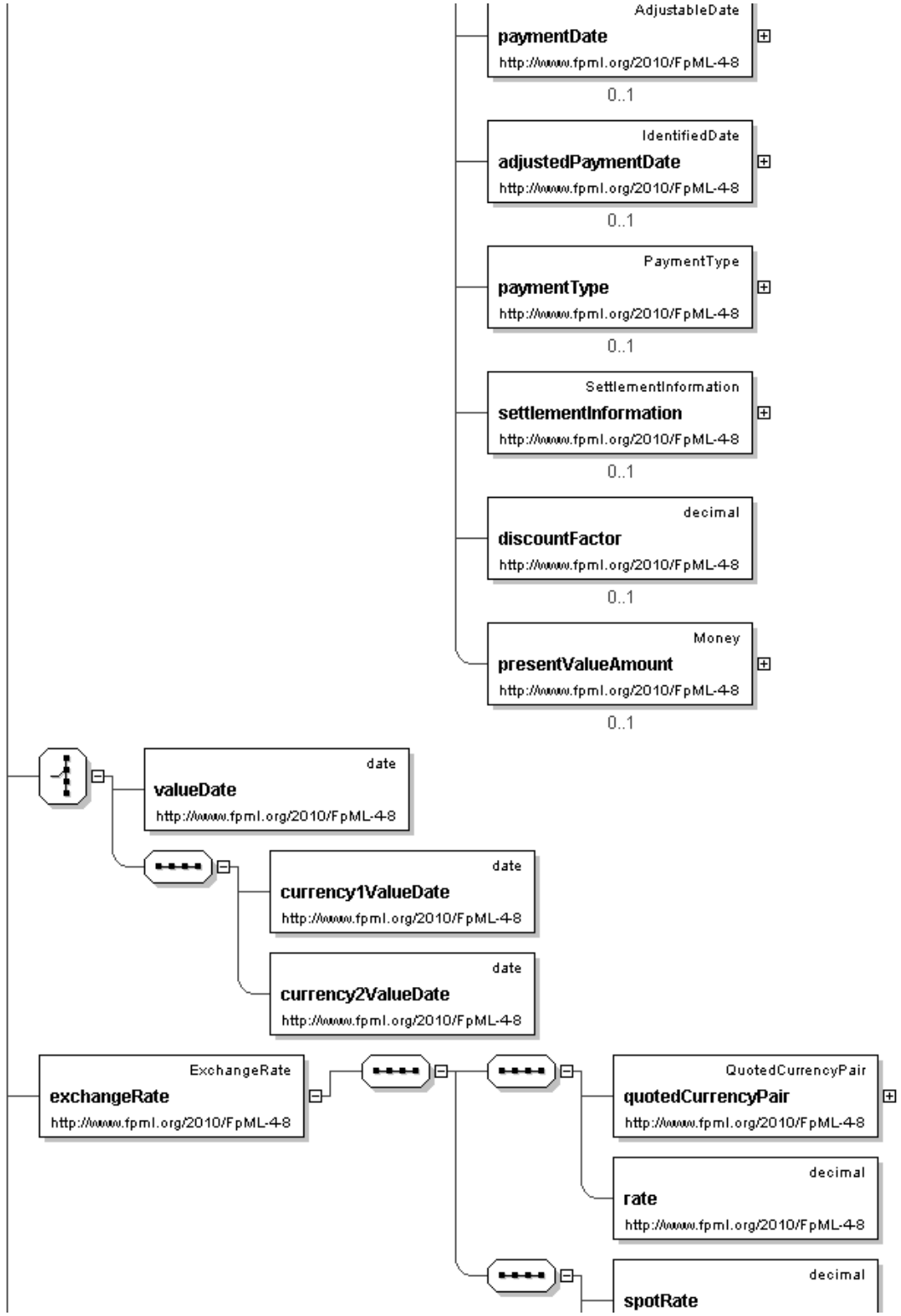
- *This element can be used wherever the following element is referenced:*
  - product

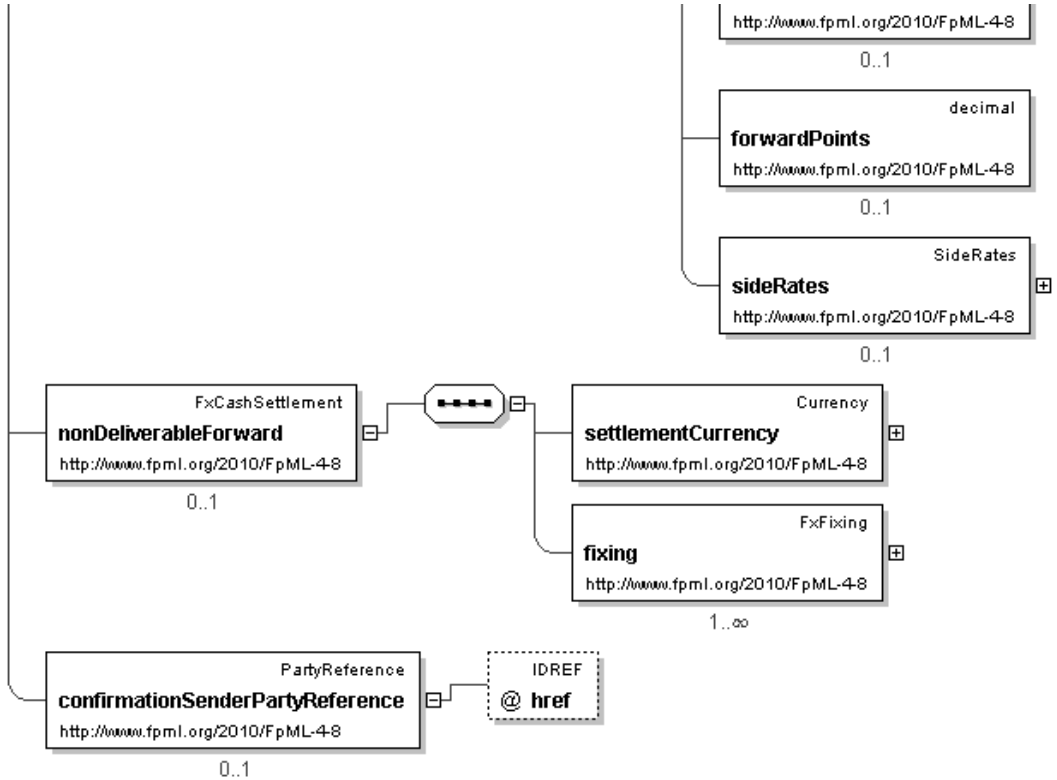
Name	fxSingleLeg
Used by (from the same schema document)	Complex Type FxSwap
Type	FxLeg
Nillable	no
Abstract	no
Documentation	A single-legged FX transaction definition (e.g., spot or forward).

Logical Diagram




paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
adjustedPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PaymentType
paymentType
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Payment
exchangedCurrency2
http://www.fpml.org/2010/FpML-4-8
IDREF
@ href
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8


AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
adjustedPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PaymentType
paymentType
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
date
valueDate
http://www.fpml.org/2010/FpML-4-8
date
currency1ValueDate
http://www.fpml.org/2010/FpML-4-8
date
currency2ValueDate
http://www.fpml.org/2010/FpML-4-8
ExchangeRate
exchangeRate
http://www.fpml.org/2010/FpML-4-8
QuotedCurrencyPair
quotedCurrencyPair
http://www.fpml.org/2010/FpML-4-8
decimal
rate
http://www.fpml.org/2010/FpML-4-8
decimal
spotRate

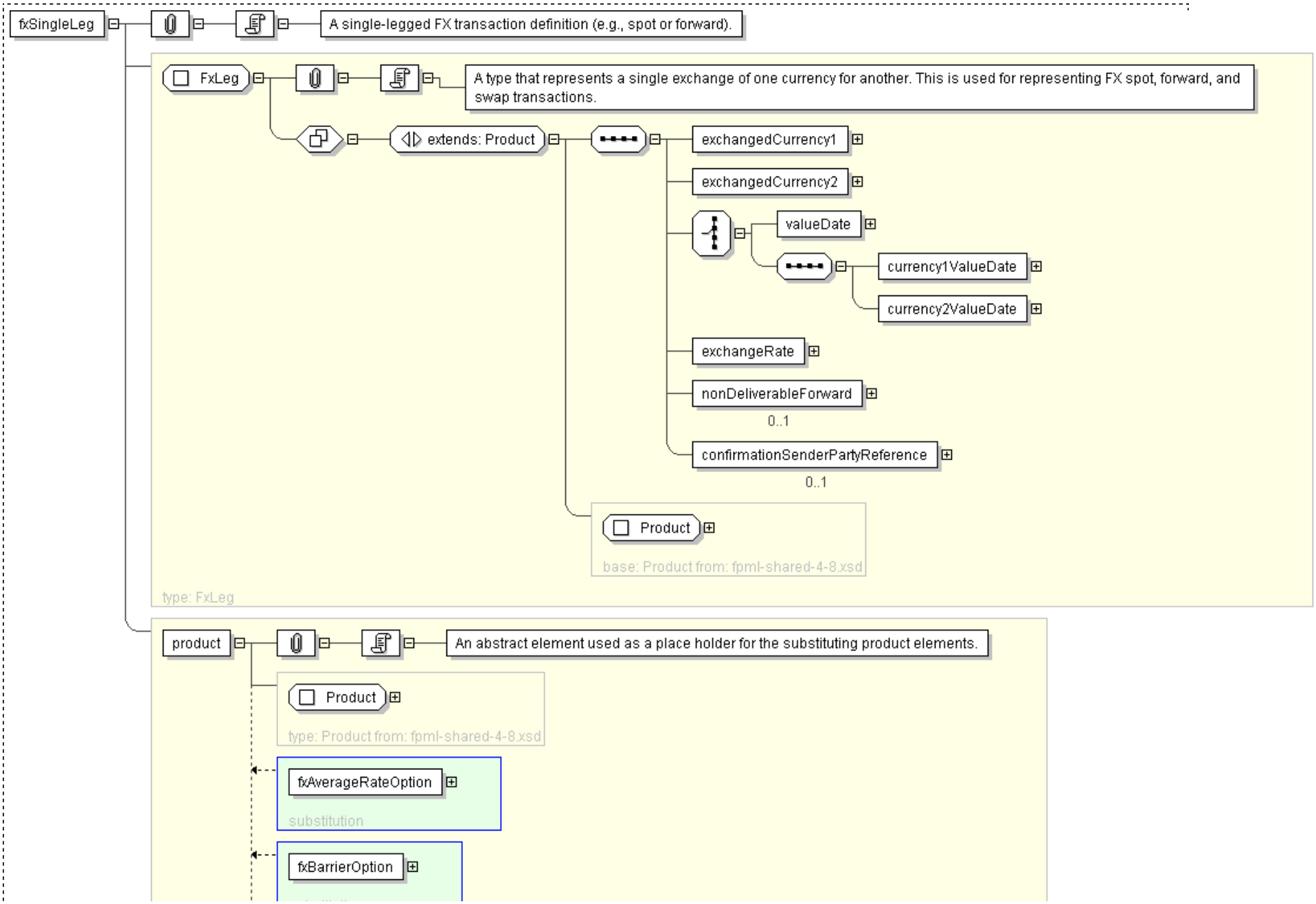


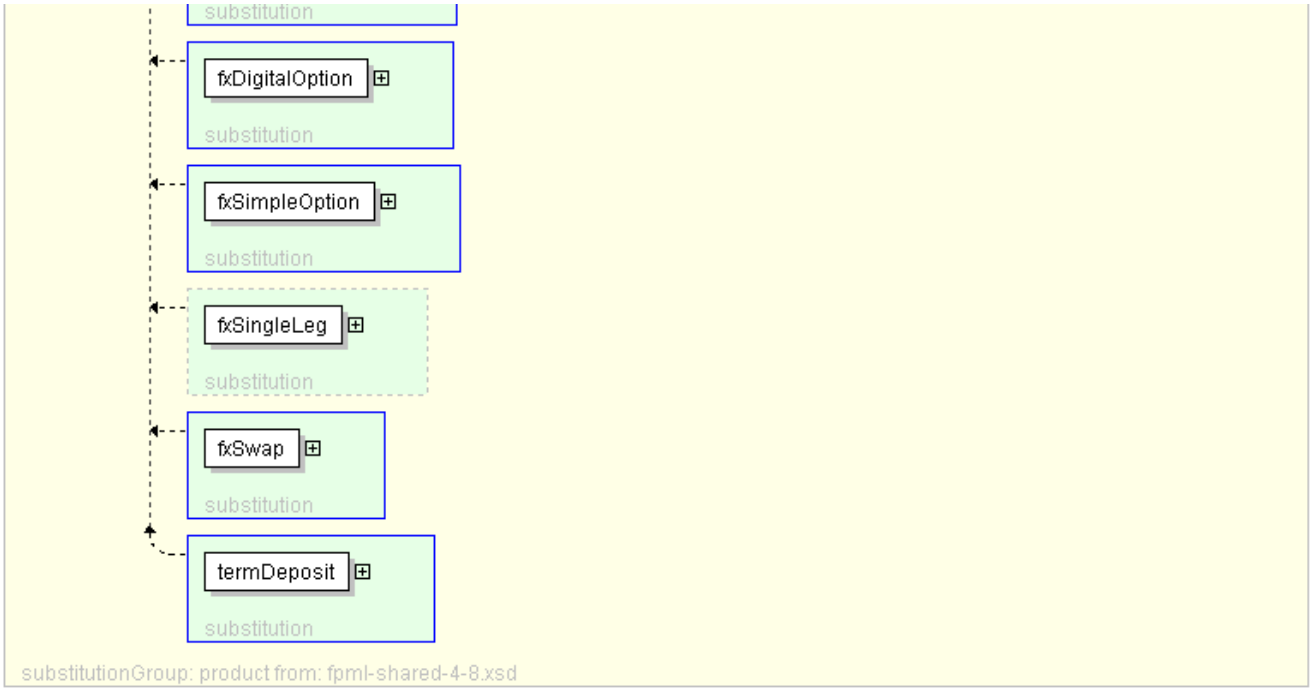
XML Instance Representation

```
<fxSingleLeg
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this
   element. Note that the domain values for this element are not strictly an enumerated list.'

   <exchangedCurrency1> Payment </exchangedCurrency1> [1]
   'This is the first of the two currency flows that define a single leg of a standard foreign exchange transaction.'

   <exchangedCurrency2> Payment </exchangedCurrency2> [1]
   'This is the second of the two currency flows that define a single leg of a standard foreign exchange transaction.'

   Start Choice [1]
      <valueDate> xsd:date </valueDate> [1]
      'The date on which both currencies traded will settle.'

      <currency1ValueDate> xsd:date </currency1ValueDate> [1]
      'The date on which the currency1 amount will be settled. To be used in a split value date scenario.'

      <currency2ValueDate> xsd:date </currency2ValueDate> [1]
      'The date on which the currency2 amount will be settled. To be used in a split value date scenario.'

   End Choice
   <exchangeRate> ExchangeRate </exchangeRate> [1]
   'The rate of exchange between the two currencies.'
```

```
<nonDeliverableForward> FxCashSettlement </nonDeliverableForward> [0..1]
'Used to describe a particular type of FX forward transaction that is settled in a single currency.'

<confirmationSenderPartyReference> PartyReference </confirmationSenderPartyReference> [0..1]
'A reference to the party that is sending the current document as a confirmation of the trade.'

</fxSingleLeg>
```

Diagram





Schema Component Representation

```
<xsd:element name="fxSingleLeg" type=" FxLeg " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

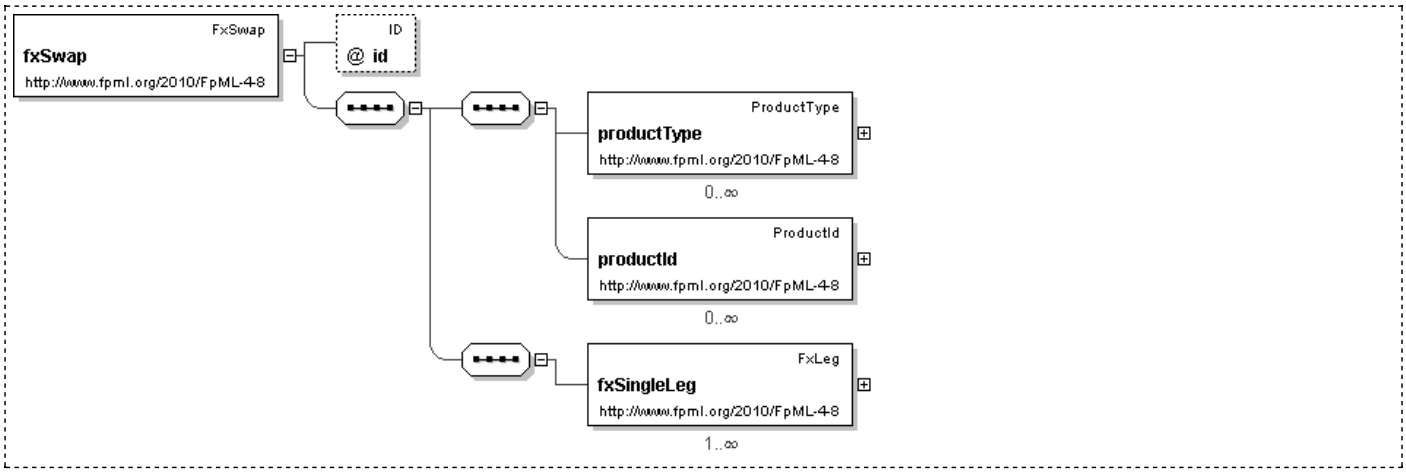
Element: fxSwap

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	fxSwap
Type	FxSwap
Nillable	no
Abstract	no
Documentation	A component describing a FX Swap product.

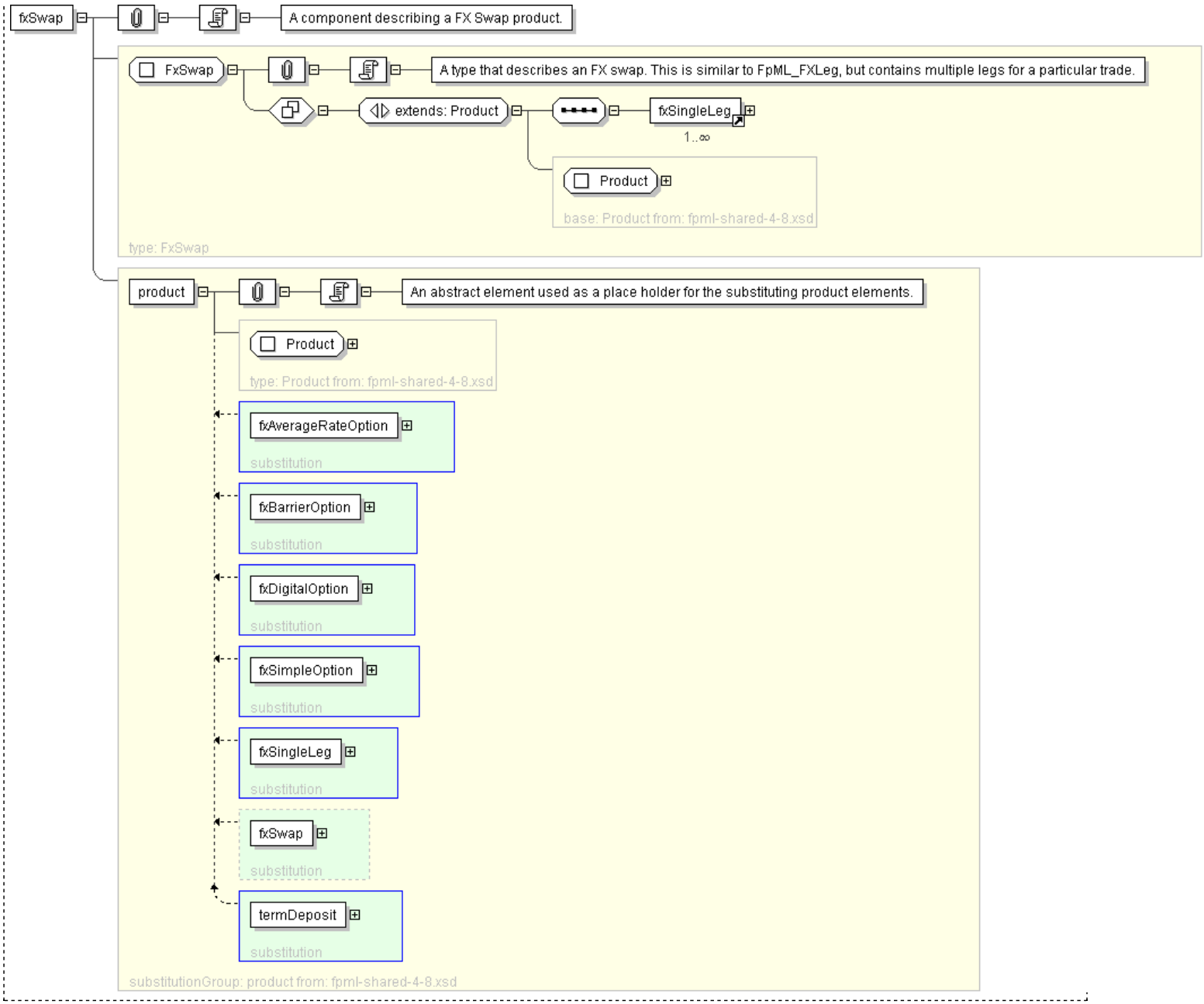
Logical Diagram



XML Instance Representation

```
<fxSwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <fxSingleLeg> ... </fxSingleLeg> [1..*]
</fxSwap>
```

Diagram



Schema Component Representation

```
<xsd:element name="fxSwap" type=" FxSwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

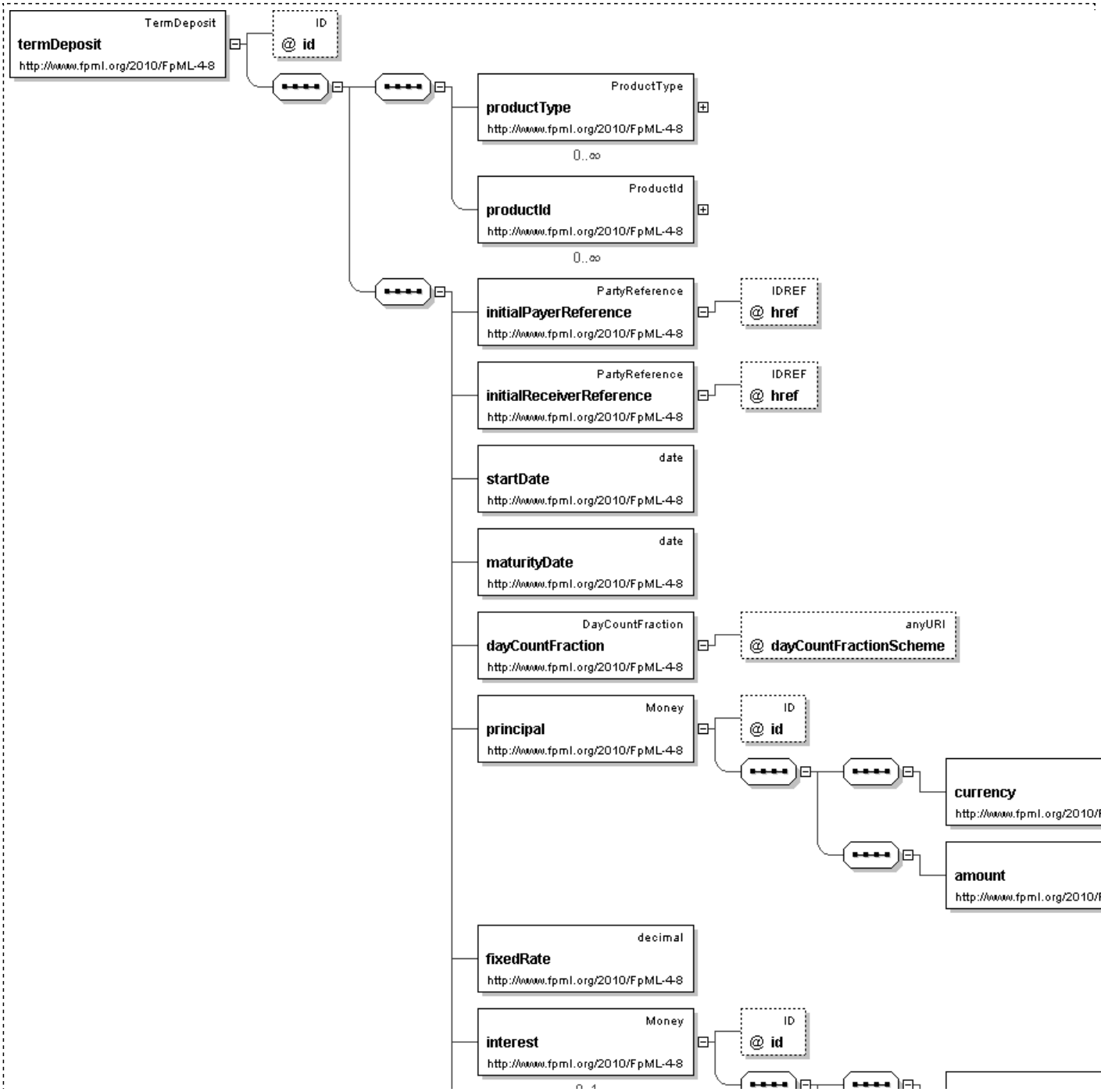
Element: termDeposit

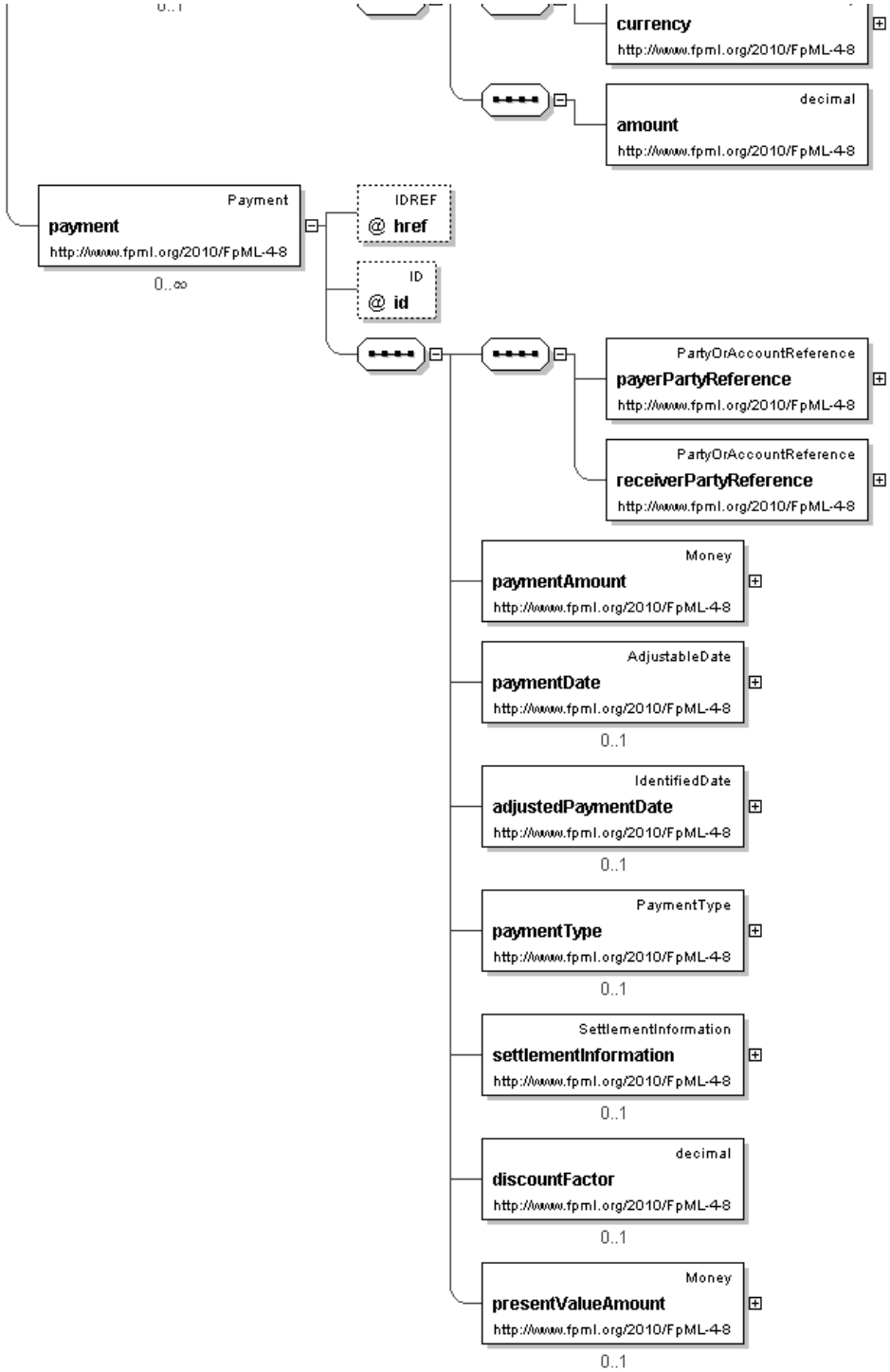
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	termDeposit
Type	TermDeposit
Nillable	no
Abstract	no
Documentation	A term deposit product definition.

Logical Diagram



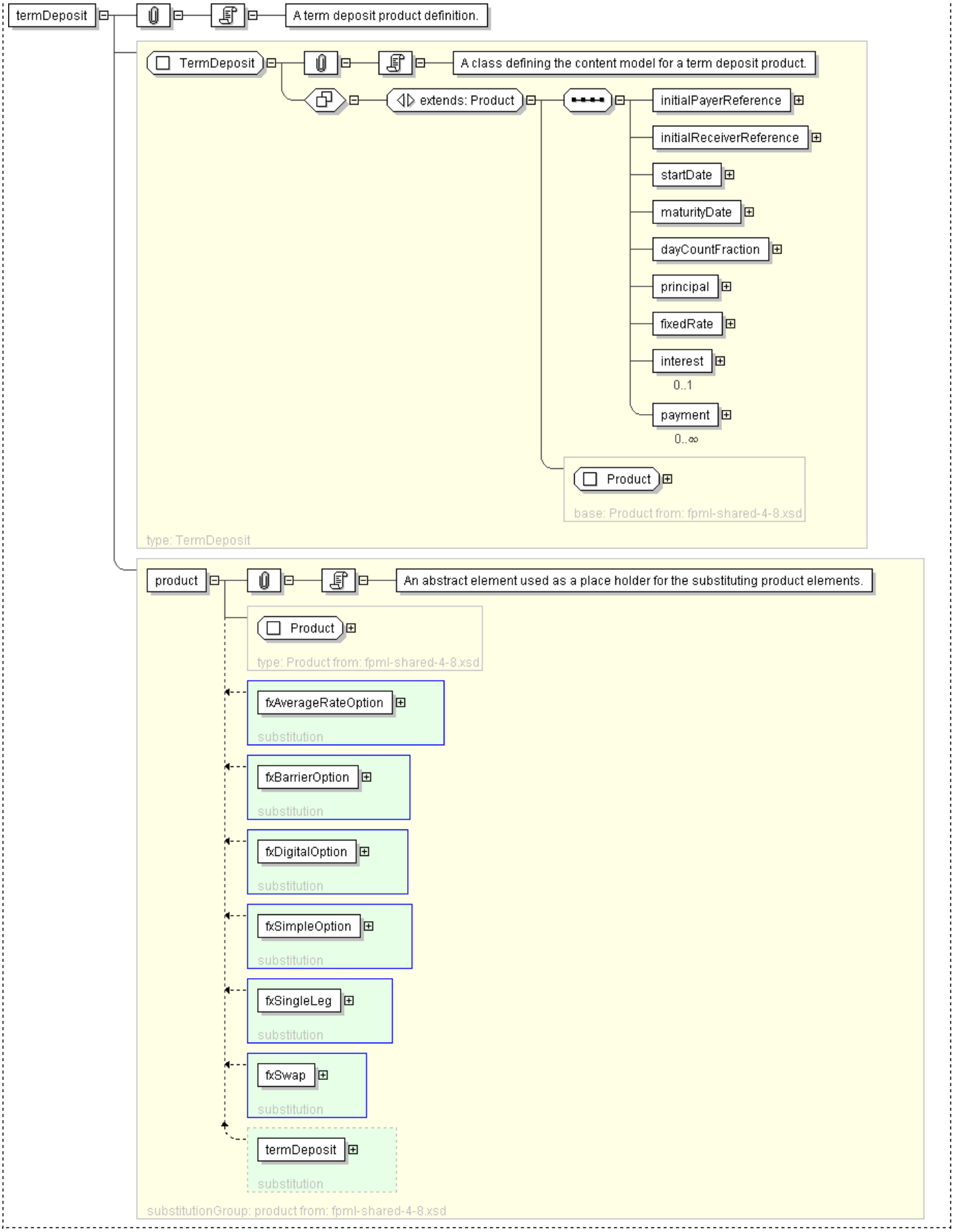


XML Instance Representation

```
<termDeposit
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated
   with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <initialPayerReference> PartyReference </initialPayerReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced is
   the payer of the initial principal of the deposit on the start date.'

   <initialReceiverReference> PartyReference </initialReceiverReference> [1]
   'A pointer style reference to a party identifier defined elsewhere in the document. The party is the
   receiver of the initial principal of the deposit on the start date.'

   <startDate> xsd:date </startDate> [1]
   'The averaging period start date.'
```

```
    <maturityDate> xsd:date </maturityDate> [1]
    'The end date of the calculation period. This date should already be adjusted for any applicable business
    day convention.'

    <dayCountFraction> DayCountFraction </dayCountFraction> [1]
    'The day count fraction.'

    <principal> Money </principal> [1]
    'The principal amount of the trade.'

    <fixedRate> xsd:decimal </fixedRate> [1]
    'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be
    represented as 0.05.'

    <interest> Money </interest> [0..1]
    'The total interest of at maturity of the trade.'

    <payment> Payment </payment> [0..*]
    'A known payment between two parties.'

</termDeposit>
```

Diagram



Schema Component Representation

```
<xsd:element name="termDeposit" type=" TermDeposit " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CutName

[Table of contents]

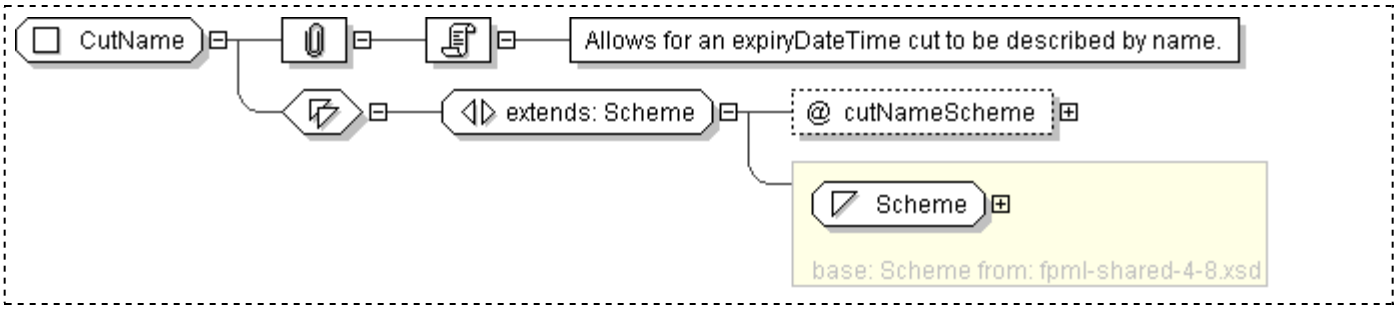
Super-types:	Scheme < CutName (by extension)
Sub-types:	None

Name	CutName
Used by (from the same schema document)	Complex Type ExpiryDateTime
Abstract	no
Documentation	Allows for an expiryDateTime cut to be described by name.

XML Instance Representation

```
<...
 cutNameScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CutName">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="cutNameScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/cut-name"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeRate

[Table of contents]

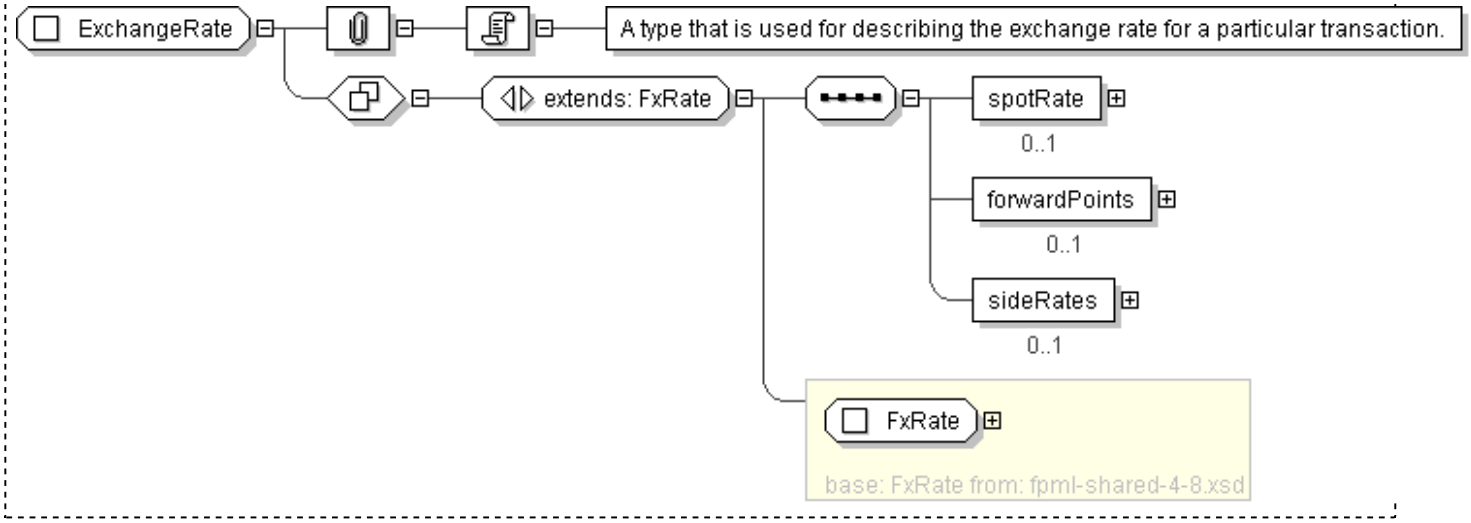
Super-types:	FxRate < ExchangeRate (by extension)
Sub-types:	None

Name	ExchangeRate
Used by (from the same schema document)	Complex Type FxLeg
Abstract	no
Documentation	A type that is used for describing the exchange rate for a particular transaction.

XML Instance Representation

```
<...>
   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between
   the two currencies.'

   <rate> xsd:decimal </rate> [1]
   'The rate of exchange between the two currencies of the leg of a deal. Must be
   specified with a quote basis.'

   <spotRate> xsd:decimal </spotRate> [0..1]
   'An optional element used for FX forwards and certain types of FX OTC options. For
   deals consumated in the FX Forwards Market, this represents the current market
   rate for a particular currency pair. For barrier and digital/binary options, it
   can be useful to include the spot rate at the time the option was executed to make
   it easier to know whether the option needs to move \"up\" or \"down\" to be
   triggered.'

   <forwardPoints> xsd:decimal </forwardPoints> [0..1]
   'An optional element used for deals consumated in the FX Forwards market. Forward
   points represent the interest rate differential between the two currencies traded
   and are quoted as a preminum or a discount. Forward points are added to, or
   subtracted from, the spot rate to create the rate of the forward trade.'

   <sideRates> SideRates </sideRates> [0..1]
   'An optional element that allow for definition of rates against base currency for
   non-base currency FX contracts.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeRate">
  <xsd:complexContent>
    <xsd:extension base=" FxRate ">
      <xsd:sequence>
        <xsd:element name="spotRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="forwardPoints" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="sideRates" type=" SideRates " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExpiryDateTime

[Table of contents]

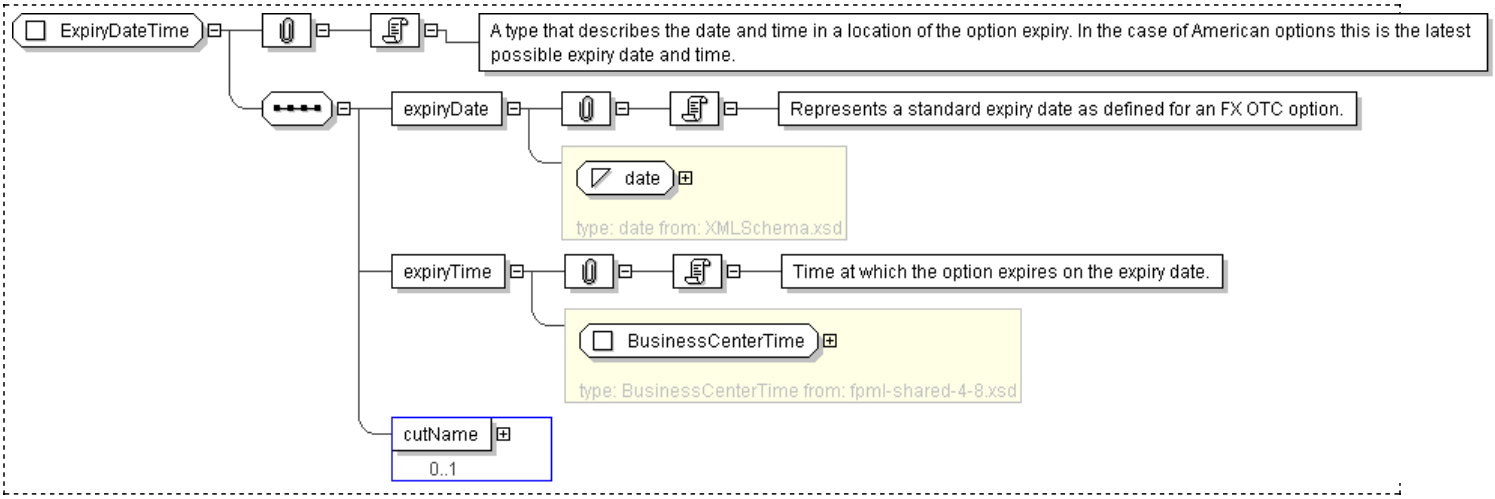
Super-types:	None
Sub-types:	None

Name	ExpiryDateTime
Used by (from the same schema document)	Complex Type ExAverageRateOption , Complex Type ExDigitalOption , Complex Type ExOptionLeg
Abstract	no
Documentation	A type that describes the date and time in a location of the option expiry. In the case of American options this is the latest possible expiry date and time.

XML Instance Representation

```
<...>
  <expiryDate> xsd:date </expiryDate> [1]
  'Represents a standard expiry date as defined for an FX OTC option.'

  <expiryTime> BusinessCenterTime </expiryTime> [1]
  'Time at which the option expires on the expiry date.'

  <cutName> CutName </cutName> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExpiryDateTime">
  <xsd:sequence>
    <xsd:element name="expiryDate" type=" xsd:date "/>
    <xsd:element name="expiryTime" type=" BusinessCenterTime "/>
    <xsd:element name="cutName" type=" CutName " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxAmericanTrigger

[Table of contents]

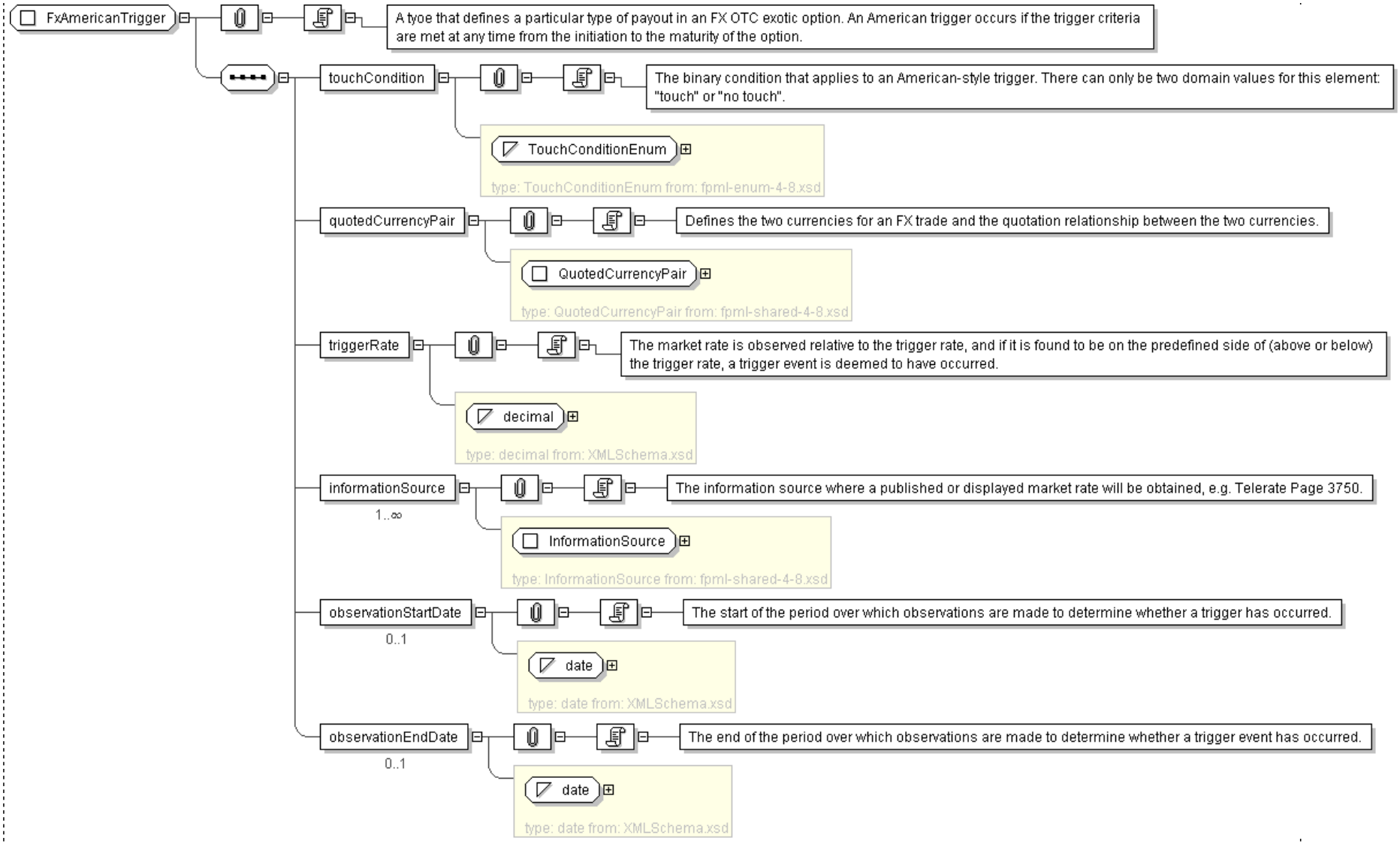
Super-types:	None
Sub-types:	None

Name	FxAmericanTrigger
Used by (from the same schema document)	Complex Type FxDigitalOption
Abstract	no
Documentation	A tyoe that defines a particular type of payout in an FX OTC exotic option. An American trigger occurs if the trigger criteria are met at any time from the initiation to the maturity of the option.

XML Instance Representation

```
<...>
   <touchCondition> TouchConditionEnum </touchCondition> [1]
   'The binary condition that applies to an American-style trigger. There can only be two domain values for this element: \"touch\" or
   \"no touch\".'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <triggerRate> xsd:decimal </triggerRate> [1]
   'The market rate is observed relative to the trigger rate, and if it is found to be on the predefined side of (above or below) the
   trigger rate, a trigger event is deemed to have occurred.'

   <informationSource> InformationSource </informationSource> [1..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

   <observationStartDate> xsd:date </observationStartDate> [0..1]
   'The start of the period over which observations are made to determine whether a trigger has occurred.'

   <observationEndDate> xsd:date </observationEndDate> [0..1]
   'The end of the period over which observations are made to determine whether a trigger event has occurred.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxAmericanTrigger">
  <xsd:sequence>
    <xsd:element name="touchCondition" type=" TouchConditionEnum "/>
    <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
    <xsd:element name="triggerRate" type=" xsd:decimal "/>
    <xsd:element name="informationSource" type=" InformationSource " maxOccurs="unbounded"/>
    <xsd:element name="observationStartDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="observationEndDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxAverageRateObservationDate

[Table of contents]

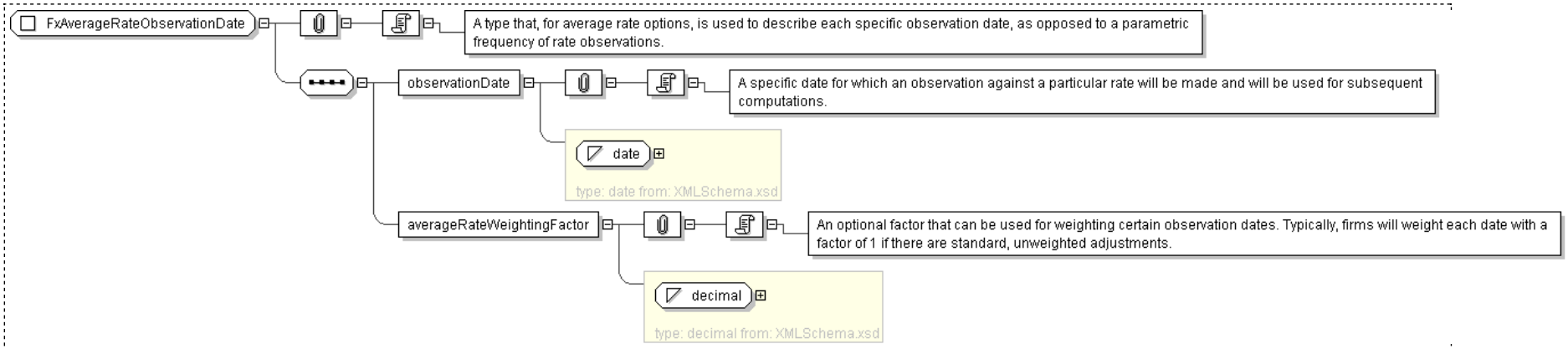
Super-types:	None
Sub-types:	None

Name	FxAverageRateObservationDate
Used by (from the same schema document)	Complex Type FxAverageRateOption
Abstract	no
Documentation	A type that, for average rate options, is used to describe each specific observation date, as opposed to a parametric frequency of rate observations.

XML Instance Representation

```
<...>
   <observationDate> xsd:date </observationDate> [1]
    'A specific date for which an observation against a particular rate will be made and will be used for subsequent computations.'

   <averageRateWeightingFactor> xsd:decimal </averageRateWeightingFactor> [1]
    'An optional factor that can be used for weighting certain observation dates. Typically, firms will weight each date with a factor of 1 if there are
    standard, unweighted adjustments.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxAverageRateObservationDate">
   <xsd:sequence>
      <xsd:element name="observationDate" type=" xsd:date "/>
      <xsd:element name="averageRateWeightingFactor" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxAverageRateObservationSchedule

[Table of contents]

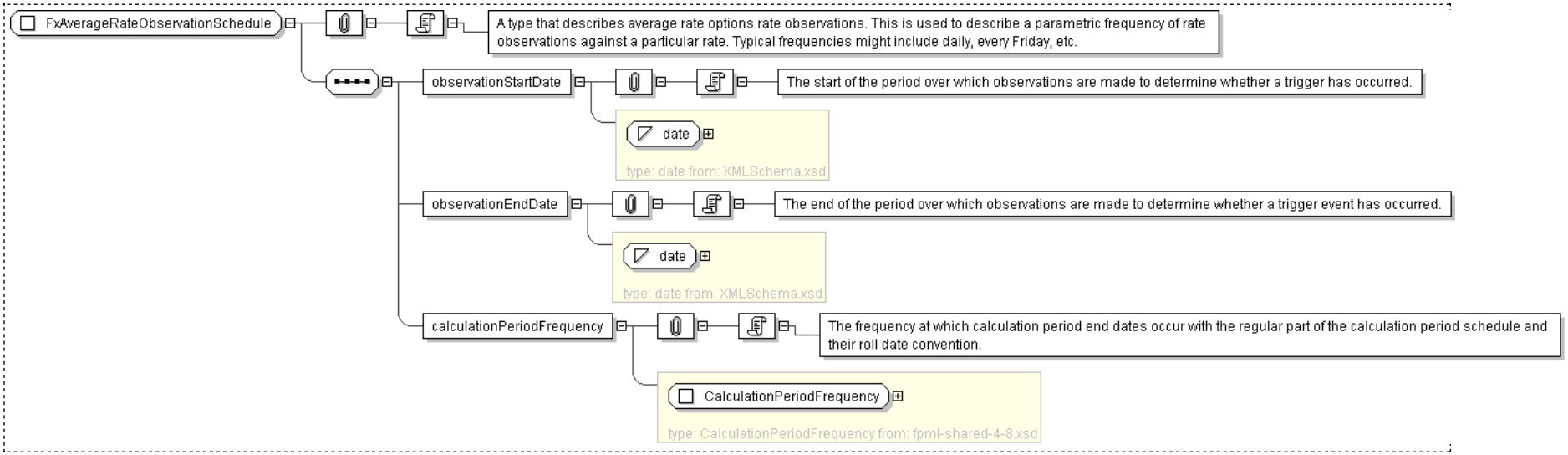
Super-types:	None
Sub-types:	None

Name	FxAverageRateObservationSchedule
Used by (from the same schema document)	Complex Type FxAverageRateOption
Abstract	no
Documentation	A type that describes average rate options rate observations. This is used to describe a parametric frequency of rate observations against a particular rate. Typical frequencies might include daily, every Friday, etc.

XML Instance Representation

```
<...>
   <observationStartDate> xsd:date </observationStartDate> [1]
    'The start of the period over which observations are made to determine whether a trigger has occurred.'

   <observationEndDate> xsd:date </observationEndDate> [1]
    'The end of the period over which observations are made to determine whether a trigger event has occurred.'

   <calculationPeriodFrequency> CalculationPeriodFrequency </calculationPeriodFrequency> [1]
    'The frequency at which calculation period end dates occur with the regular part of the calculation period schedule and their roll date convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxAverageRateObservationSchedule">
  <xsd:sequence>
    <xsd:element name="observationStartDate" type=" xsd:date "/>
    <xsd:element name="observationEndDate" type=" xsd:date "/>
    <xsd:element name="calculationPeriodFrequency" type=" CalculationPeriodFrequency "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxAverageRateOption

[Table of contents]

Super-types:	Product < FxAverageRateOption (by extension)
Sub-types:	None

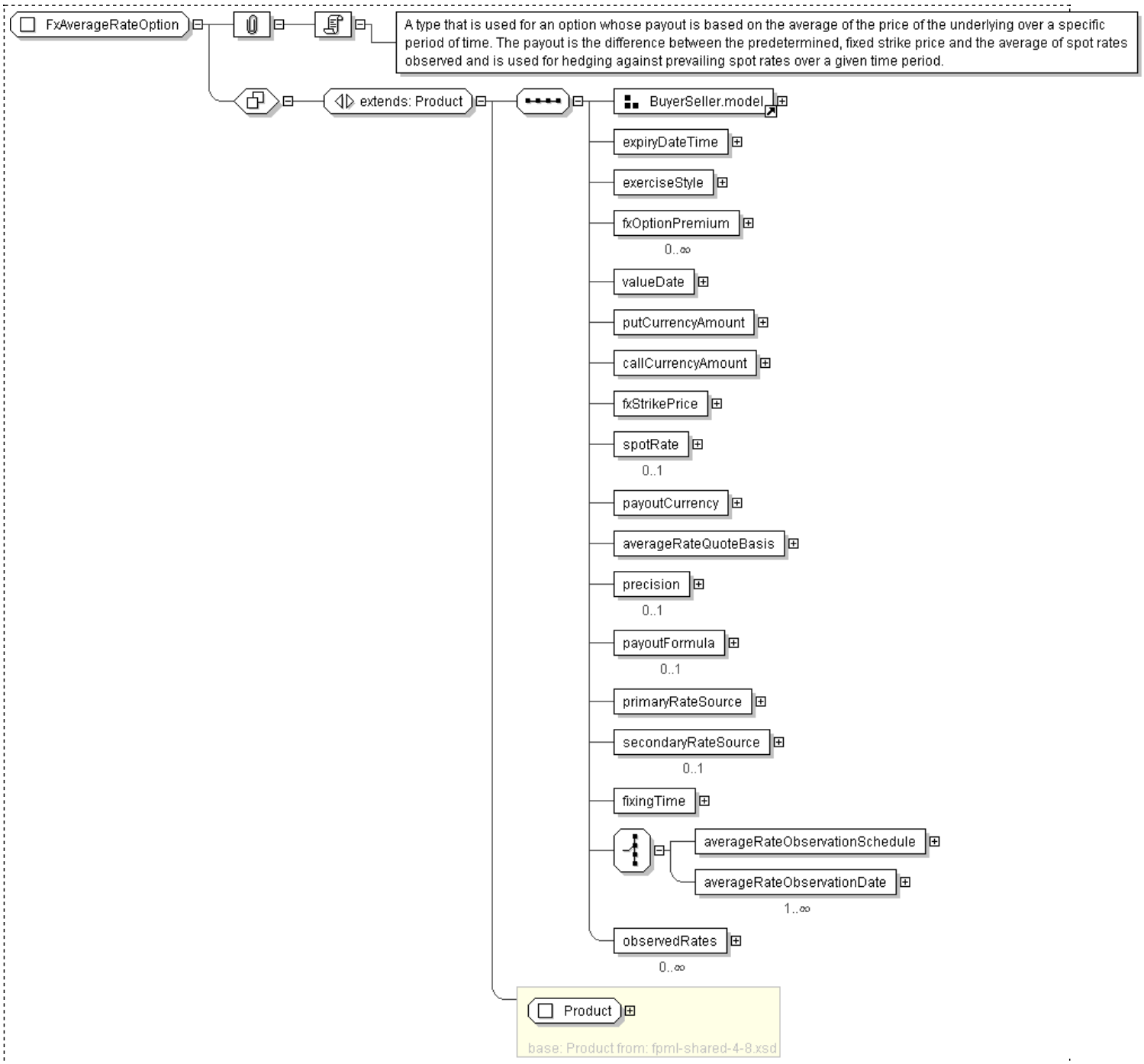
Name	FxAverageRateOption
Used by (from the same schema document)	Element fxAverageRateOption
Abstract	no
Documentation	A type that is used for an option whose payout is based on the average of the price of the underlying over a specific period of time. The payout is the difference between the predetermined, fixed strike price and the average of spot rates observed and is used for hedging against prevailing spot rates over a given time period.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this
   instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of
   FRAs this is the floating rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
   'The date and time in a location of the option expiry. In the case of american options this is the latest
   possible expiry date and time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
   'The manner in which the option can be exercised.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
   'Premium amount or premium installment amount for an option.'

   <valueDate> xsd:date </valueDate> [1]
   'The date on which both currencies traded will settle.'

   <putCurrencyAmount> Money </putCurrencyAmount> [1]
   'The currency amount that the option gives the right to sell.'

   <callCurrencyAmount> Money </callCurrencyAmount> [1]
   'The currency amount that the option gives the right to buy.'

   <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
   'TBA'

   <spotRate> xsd:decimal </spotRate> [0..1]
   'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX
   Forwards Market, this represents the current market rate for a particular currency pair. For barrier and
   digital/binary options, it can be useful to include the spot rate at the time the option was executed to make
   it easier to know whether the option needs to move \"up\" or \"down\" to be triggered.'

   <payoutCurrency> Currency </payoutCurrency> [1]
   'The ISO code of the currency in which a payout (if any) is to be made when a trigger is hit on a digital or
   barrier option.'

   <averageRateQuoteBasis> StrikeQuoteBasisEnum </averageRateQuoteBasis> [1]
   'The method by which the average rate that is being observed is quoted.'

   <precision> xsd:nonNegativeInteger </precision> [0..1]
   'Specifies the rounding precision in terms of a number of decimal places. Note how a percentage rate rounding
   of 5 decimal places is expressed as a rounding precision of 7 in the FpML document since the percentage is
   expressed as a decimal, e.g. 9.876543% (or 0.09876543) being rounded to the nearest 5 decimal places is
   9.87654% (or 0.0987654).'

   <payoutFormula> xsd:string </payoutFormula> [0..1]
   'The description of the mathematical computation for how the payout is computed.'

   <primaryRateSource> InformationSource </primaryRateSource> [1]
   'The primary source for where the rate observation will occur. Will typically be either a page or a reference
   bank published rate.'
```

```
<secondaryRateSource> InformationSource </secondaryRateSource> [0..1]
'An alternative, or secondary, source for where the rate observation will occur. Will typically be either a page or a reference bank published rate.'

<fixingTime> BusinessCenterTime </fixingTime> [1]
'The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g. 11:00am London time.'

Start Choice [1]
   <averageRateObservationSchedule> FxAverageRateObservationSchedule </averageRateObservationSchedule> [1]
   'Parametric schedule of rate observations.'

   <averageRateObservationDate> FxAverageRateObservationDate </averageRateObservationDate> [1..*]
   'One of more specific rate observation dates.'

End Choice
<observedRates> ObservedRates </observedRates> [0..*]
'Describes prior rate observations within average rate options. Periodically, an average rate option agreement will be struck whereby some rates have already been observed in the past but will become part of computation of the average rate of the option. This structure provides for these previously observed rates to be included in the description of the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxAverageRateOption">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model "/>
        <xsd:element name="expiryDateTime" type=" ExpiryDateTime "/>
        <xsd:element name="exerciseStyle" type=" ExerciseStyleEnum "/>
        <xsd:element name="fxOptionPremium" type=" FxOptionPremium " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="valueDate" type=" xsd:date "/>
        <xsd:element name="putCurrencyAmount" type=" Money "/>
        <xsd:element name="callCurrencyAmount" type=" Money "/>
        <xsd:element name="fxStrikePrice" type=" FxStrikePrice "/>
        <xsd:element name="spotRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="payoutCurrency" type=" Currency "/>
        <xsd:element name="averageRateQuoteBasis" type=" StrikeQuoteBasisEnum "/>
        <xsd:element name="precision" type=" xsd:nonNegativeInteger " minOccurs="0"/>
        <xsd:element name="payoutFormula" type=" xsd:string " minOccurs="0"/>
        <xsd:element name="primaryRateSource" type=" InformationSource "/>
        <xsd:element name="secondaryRateSource" type=" InformationSource " minOccurs="0"/>
        <xsd:element name="fixingTime" type=" BusinessCenterTime "/>
        <xsd:choice>
          <xsd:element name="averageRateObservationSchedule" type=" FxAverageRateObservationSchedule "/>
          <xsd:element name="averageRateObservationDate" type=" FxAverageRateObservationDate "
          maxOccurs="unbounded"/>
        </xsd:choice>
        <xsd:element name="observedRates" type=" ObservedRates " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxBarrier

[Table of contents]

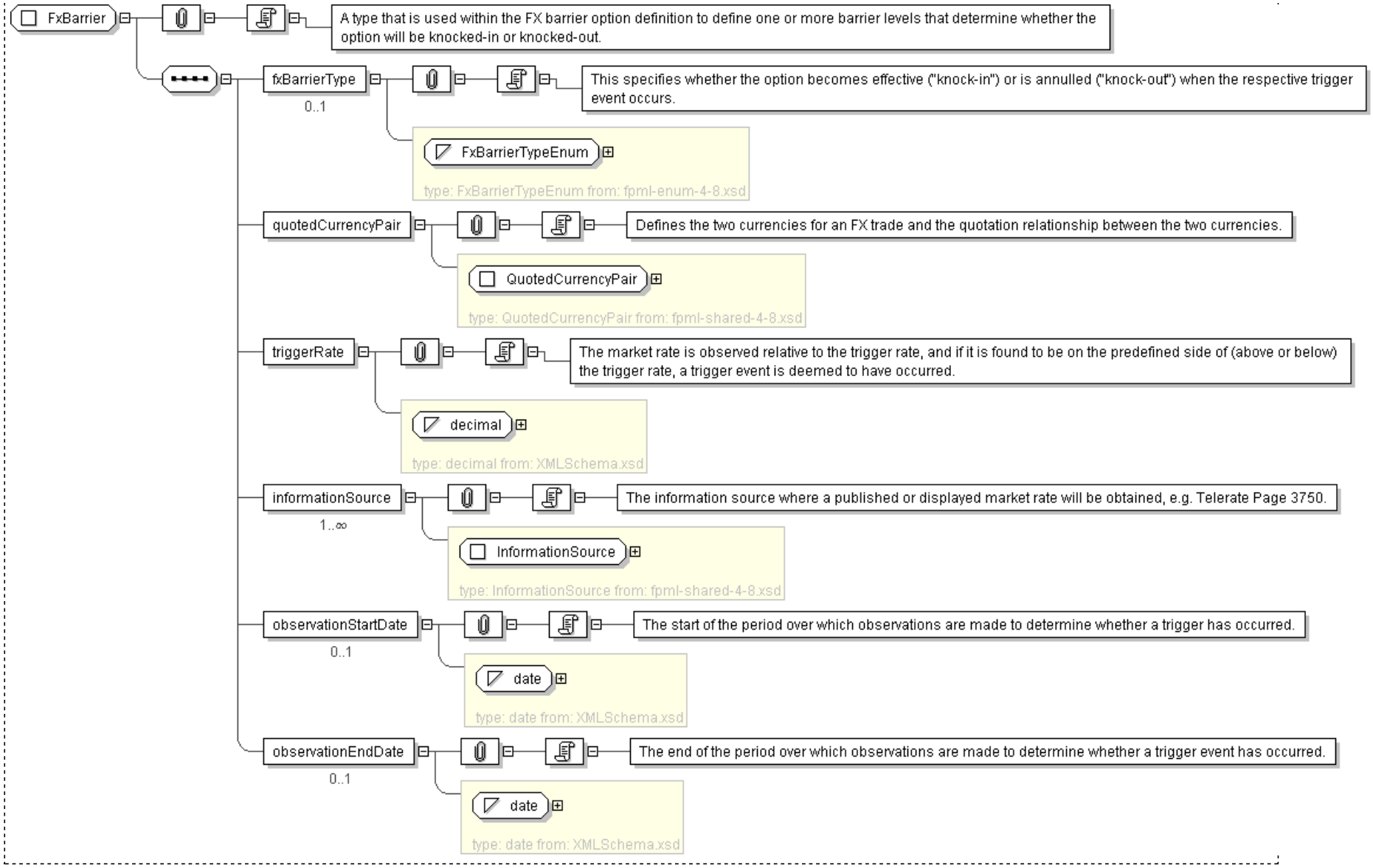
Super-types:	None
Sub-types:	None

Name	FxBarrier
Used by (from the same schema document)	Complex Type FxBarrierOption
Abstract	no
Documentation	A type that is used within the FX barrier option definition to define one or more barrier levels that determine whether the option will be knocked-in or knocked-out.

XML Instance Representation

```
<...>
   <fxBarrierType> FxBarrierTypeEnum </fxBarrierType> [0..1]
   'This specifies whether the option becomes effective (\"knock-in\") or is annulled (\"knock-out\") when the respective trigger
   event occurs.'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <triggerRate> xsd:decimal </triggerRate> [1]
   'The market rate is observed relative to the trigger rate, and if it is found to be on the predefined side of (above or below)
   the trigger rate, a trigger event is deemed to have occurred.'

   <informationSource> InformationSource </informationSource> [1..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

   <observationStartDate> xsd:date </observationStartDate> [0..1]
   'The start of the period over which observations are made to determine whether a trigger has occurred.'

   <observationEndDate> xsd:date </observationEndDate> [0..1]
   'The end of the period over which observations are made to determine whether a trigger event has occurred.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxBarrier">
  <xsd:sequence>
    <xsd:element name="fxBarrierType" type=" FxBarrierTypeEnum " minOccurs="0"/>
    <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
    <xsd:element name="triggerRate" type=" xsd:decimal "/>
    <xsd:element name="informationSource" type=" InformationSource " maxOccurs="unbounded"/>
    <xsd:element name="observationStartDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="observationEndDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxBarrierOption

[Table of contents]

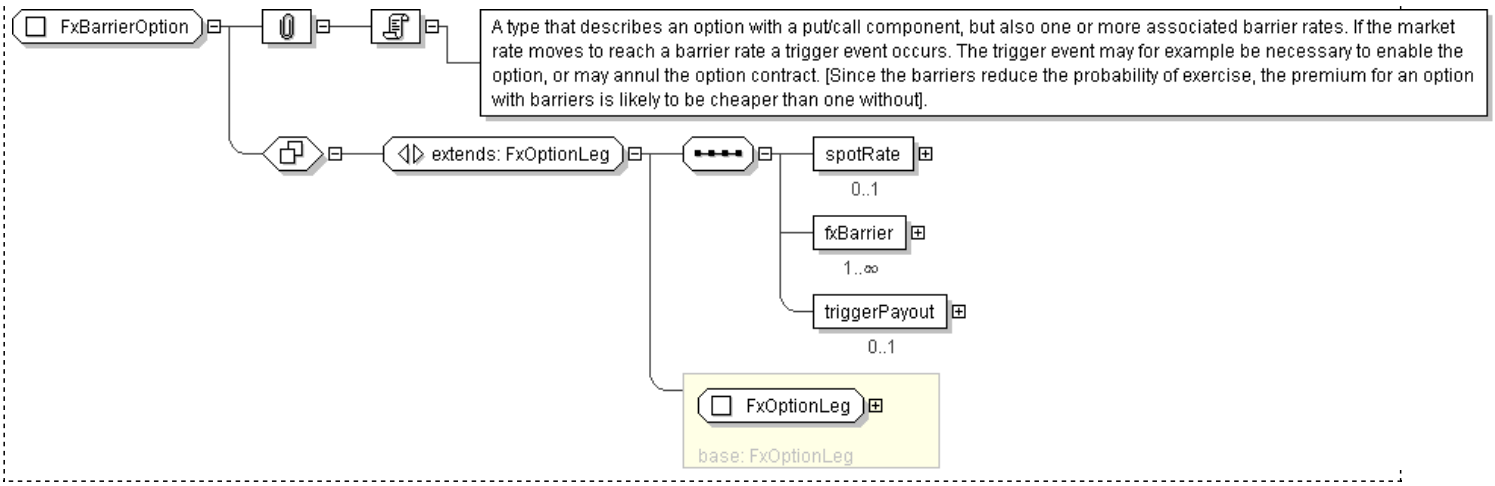
Super-types:	Product < FxOptionLeg (by extension) < FxBarrierOption (by extension)
Sub-types:	None

Name	FxBarrierOption
Used by (from the same schema document)	Element fxBarrierOption
Abstract	no
Documentation	A type that describes an option with a put/call component, but also one or more associated barrier rates. If the market rate moves to reach a barrier rate a trigger event occurs. The trigger event may for example be necessary to enable the option, or may annul the option contract. [Since the barriers reduce the probability of exercise, the premium for an option with barriers is likely to be cheaper than one without].

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the
   case of FRAs this is the floating rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
   'The date and time in a location of the option expiry. In the case of american options this is the latest
   possible expiry date and time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
   'The manner in which the option can be exercised.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
   'Premium amount or premium installment amount for an option.'

   <valueDate> xsd:date </valueDate> [1]
   'The date on which both currencies traded will settle.'

   <cashSettlementTerms> FxCashSettlement </cashSettlementTerms> [0..1]
   'This optional element is only used if an option has been specified at execution time to be settled into a
   single cash payment. This would be used for a non-deliverable option.'

   <putCurrencyAmount> Money </putCurrencyAmount> [1]
   'The currency amount that the option gives the right to sell.'

   <callCurrencyAmount> Money </callCurrencyAmount> [1]
   'The currency amount that the option gives the right to buy.'

   <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
   'TBA'

   <quotedAs> QuotedAs </quotedAs> [0..1]
   'Describes how the option was quoted.'

   <spotRate> xsd:decimal </spotRate> [0..1]
   'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the
   FX Forwards Market, this represents the current market rate for a particular currency pair. For barrier and
   digital/binary options, it can be useful to include the spot rate at the time the option was executed to
   make it easier to know whether the option needs to move \"up\" or \"down\" to be triggered.'

   <fxBarrier> FxBarrier </fxBarrier> [1..*]
   'Information about a barrier rate in a Barrier Option - specifying the exact criteria for a trigger event to
   occur.'

   <triggerPayout> FxOptionPayout </triggerPayout> [0..1]
   'The amount of currency which becomes payable if and when a trigger event occurs.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxBarrierOption">
  <xsd:complexContent>
    <xsd:extension base=" FxOptionLeg ">
      <xsd:sequence>
        <xsd:element name="spotRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="fxBarrier" type=" FxBarrier " maxOccurs="unbounded"/>
        <xsd:element name="triggerPayout" type=" FxOptionPayout " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxDigitalOption

[Table of contents]

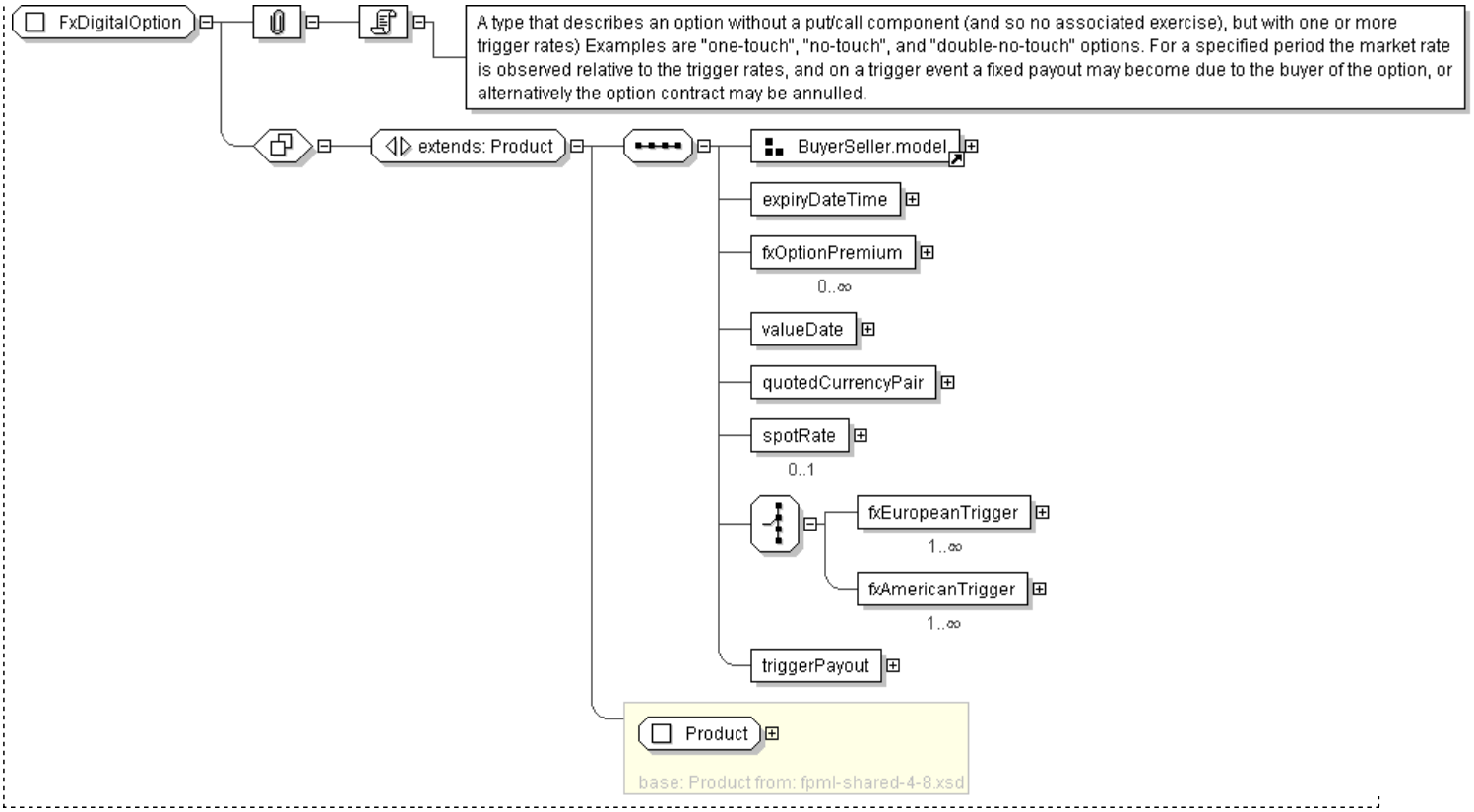
Super-types:	Product < FxDigitalOption (by extension)
Sub-types:	None

Name	FxDigitalOption
Used by (from the same schema document)	Element fxDigitalOption
Abstract	no
Documentation	A type that describes an option without a put/call component (and so no associated exercise), but with one or more trigger rates) Examples are "one-touch", "no-touch", and "double-no-touch" options. For a specified period the market rate is observed relative to the trigger rates, and on a trigger event a fixed payout may become due to the buyer of the option, or alternatively the option contract may be annulled.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the
   case of FRAs this is the floating rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
   'The date and time in a location of the option expiry. In the case of american options this is the latest
   possible expiry date and time.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
   'Premium amount or premium installment amount for an option.'

   <valueDate> xsd:date </valueDate> [1]
   'The date on which both currencies traded will settle.'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <spotRate> xsd:decimal </spotRate> [0..1]
   'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX
   Forwards Market, this represents the current market rate for a particular currency pair. For barrier and
   digital/binary options, it can be useful to include the spot rate at the time the option was executed to make
   it easier to know whether the option needs to move \"up\" or \"down\" to be triggered.'

   Start Choice [1]
      <fxEuropeanTrigger> FxEuropeanTrigger </fxEuropeanTrigger> [1..*]
      'A European trigger occurs if the trigger criteria are met, but these are valid (and an observation is
      made) only at the maturity of the option.'

      <fxAmericanTrigger> FxAmericanTrigger </fxAmericanTrigger> [1..*]
      'An American trigger occurs if the trigger criteria are met at any time from the initiation to the
      maturity of the option.'

   End Choice
   <triggerPayout> FxOptionPayout </triggerPayout> [1]
   'The amount of currency which becomes payable if and when a trigger event occurs.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxDigitalOption">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model "/>
        <xsd:element name="expiryDateTime" type=" ExpiryDateTime "/>
        <xsd:element name="fxOptionPremium" type=" FxOptionPremium " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="valueDate" type=" xsd:date "/>
        <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
        <xsd:element name="spotRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="fxEuropeanTrigger" type=" FxEuropeanTrigger " maxOccurs="unbounded"/>
          <xsd:element name="fxAmericanTrigger" type=" FxAmericanTrigger " maxOccurs="unbounded"/>
        </xsd:choice>
        <xsd:element name="triggerPayout" type=" FxOptionPayout "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxEuropeanTrigger

[Table of contents]

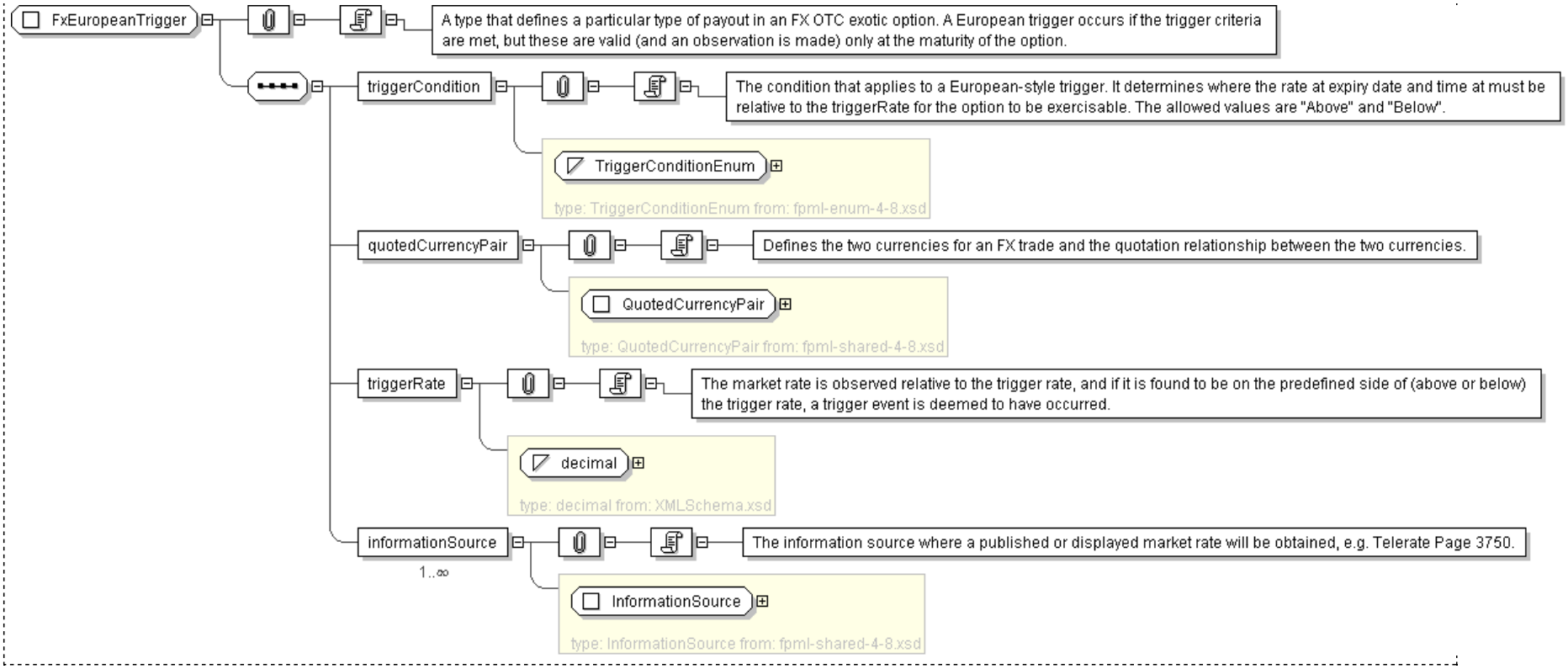
Super-types:	None
Sub-types:	None

Name	FxEuropeanTrigger
Used by (from the same schema document)	Complex Type FxDigitalOption
Abstract	no
Documentation	A type that defines a particular type of payout in an FX OTC exotic option. A European trigger occurs if the trigger criteria are met, but these are valid (and an observation is made) only at the maturity of the option.

XML Instance Representation

```
<...>
   <triggerCondition> TriggerConditionEnum </triggerCondition> [1]
   'The condition that applies to a European-style trigger. It determines where the rate at expiry date and time at must be relative to
   the triggerRate for the option to be exercisable. The allowed values are \"Above\" and \"Below\".'

   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <triggerRate> xsd:decimal </triggerRate> [1]
   'The market rate is observed relative to the trigger rate, and if it is found to be on the predefined side of (above or below) the
   trigger rate, a trigger event is deemed to have occurred.'

   <informationSource> InformationSource </informationSource> [1..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxEuropeanTrigger">
  <xsd:sequence>
    <xsd:element name="triggerCondition" type=" TriggerConditionEnum "/>
    <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
    <xsd:element name="triggerRate" type=" xsd:decimal "/>
    <xsd:element name="informationSource" type=" InformationSource " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxLeg

[Table of contents]

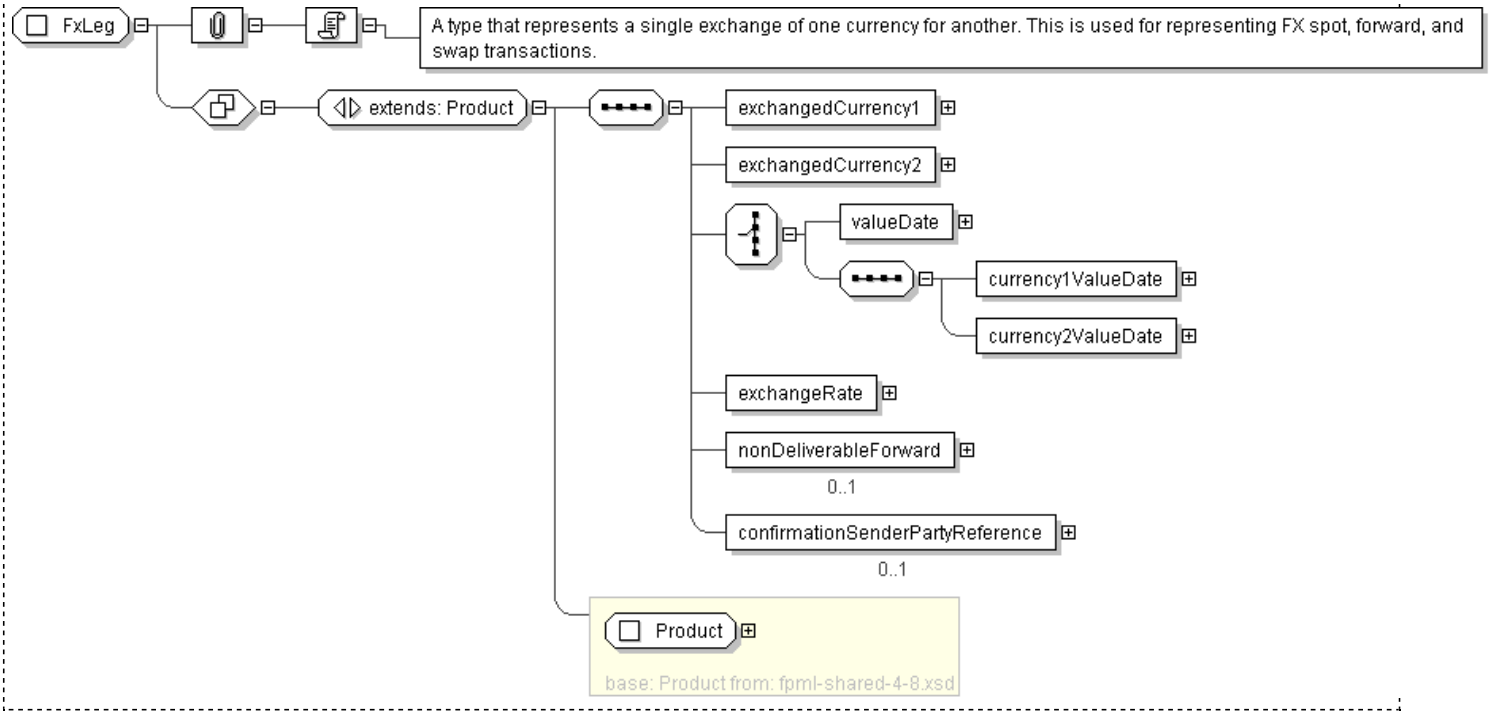
Super-types:	Product < FxLeg (by extension)
Sub-types:	None

Name	FxLeg
Used by (from the same schema document)	Element fxSingleLeg
Abstract	no
Documentation	A type that represents a single exchange of one currency for another. This is used for representing FX spot, forward, and swap transactions.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <exchangedCurrency1> Payment </exchangedCurrency1> [1]
   'This is the first of the two currency flows that define a single leg of a standard foreign exchange transaction.'

   <exchangedCurrency2> Payment </exchangedCurrency2> [1]
   'This is the second of the two currency flows that define a single leg of a standard foreign exchange transaction.'

   Start Choice [1]
      <valueDate> xsd:date </valueDate> [1]
      'The date on which both currencies traded will settle.'

      <currency1ValueDate> xsd:date </currency1ValueDate> [1]
      'The date on which the currency1 amount will be settled. To be used in a split value date scenario.'

      <currency2ValueDate> xsd:date </currency2ValueDate> [1]
      'The date on which the currency2 amount will be settled. To be used in a split value date scenario.'

   End Choice
   <exchangeRate> ExchangeRate </exchangeRate> [1]
   'The rate of exchange between the two currencies.'

   <nonDeliverableForward> FxCashSettlement </nonDeliverableForward> [0..1]
   'Used to describe a particular type of FX forward transaction that is settled in a single currency.'

   <confirmationSenderPartyReference> PartyReference </confirmationSenderPartyReference> [0..1]
   'A reference to the party that is sending the current document as a confirmation of the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxLeg">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="exchangedCurrency1" type=" Payment "/>
        <xsd:element name="exchangedCurrency2" type=" Payment "/>
        <xsd:choice>
          <xsd:element name="valueDate" type=" xsd:date "/>
          <xsd:sequence>
            <xsd:element name="currency1ValueDate" type=" xsd:date "/>
            <xsd:element name="currency2ValueDate" type=" xsd:date "/>
          </xsd:sequence>
        </xsd:choice>
        <xsd:element name="exchangeRate" type=" ExchangeRate "/>
        <xsd:element name="nonDeliverableForward" type=" FxCashSettlement " minOccurs="0"/>
        <xsd:element name="confirmationSenderPartyReference" type=" PartyReference " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxOptionLeg

[Table of contents]

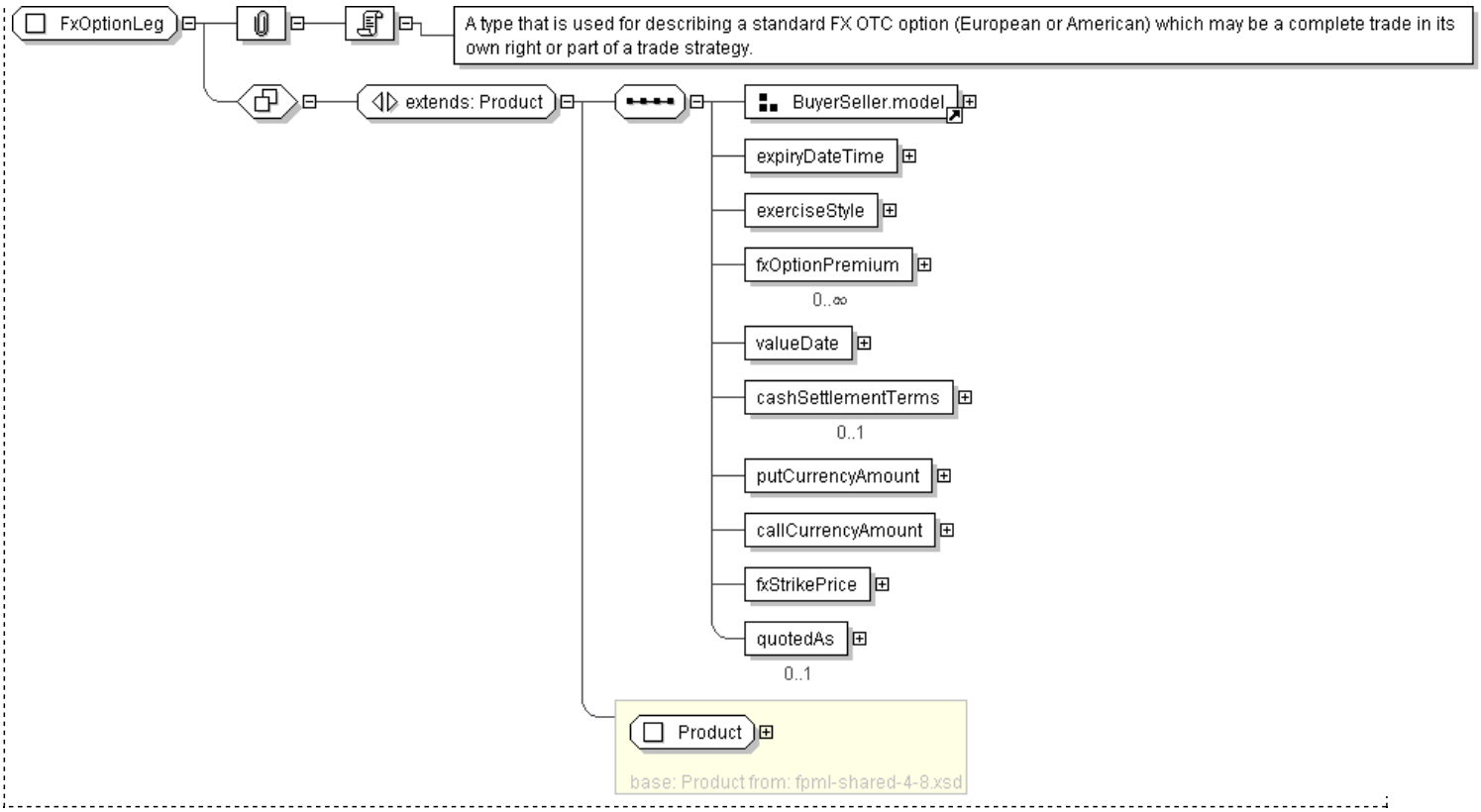
Super-types:	Product < FxOptionLeg (by extension)
Sub-types:	• FxBarrierOption (by extension)

Name	FxOptionLeg
Used by (from the same schema document)	Element fxSimpleOption
Abstract	no
Documentation	A type that is used for describing a standard FX OTC option (European or American) which may be a complete trade in its own right or part of a trade strategy.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the
   case of FRAs this is the floating rate payer.'

   <expiryDateTime> ExpiryDateTime </expiryDateTime> [1]
   'The date and time in a location of the option expiry. In the case of american options this is the latest
   possible expiry date and time.'

   <exerciseStyle> ExerciseStyleEnum </exerciseStyle> [1]
   'The manner in which the option can be exercised.'

   <fxOptionPremium> FxOptionPremium </fxOptionPremium> [0..*]
   'Premium amount or premium installment amount for an option.'

   <valueDate> xsd:date </valueDate> [1]
   'The date on which both currencies traded will settle.'

   <cashSettlementTerms> FxCashSettlement </cashSettlementTerms> [0..1]
   'This optional element is only used if an option has been specified at execution time to be settled into a
   single cash payment. This would be used for a non-deliverable option.'

   <putCurrencyAmount> Money </putCurrencyAmount> [1]
   'The currency amount that the option gives the right to sell.'

   <callCurrencyAmount> Money </callCurrencyAmount> [1]
   'The currency amount that the option gives the right to buy.'

   <fxStrikePrice> FxStrikePrice </fxStrikePrice> [1]
   'TBA'

   <quotedAs> QuotedAs </quotedAs> [0..1]
   'Describes how the option was quoted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxOptionLeg">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model "/>
        <xsd:element name="expiryDateTime" type=" ExpiryDateTime "/>
        <xsd:element name="exerciseStyle" type=" ExerciseStyleEnum "/>
        <xsd:element name="fxOptionPremium" type=" FxOptionPremium " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="valueDate" type=" xsd:date "/>
        <xsd:element name="cashSettlementTerms" type=" FxCashSettlement " minOccurs="0"/>
        <xsd:element name="putCurrencyAmount" type=" Money "/>
        <xsd:element name="callCurrencyAmount" type=" Money "/>
        <xsd:element name="fxStrikePrice" type=" FxStrikePrice "/>
        <xsd:element name="quotedAs" type=" QuotedAs " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxOptionPayout

[Table of contents]

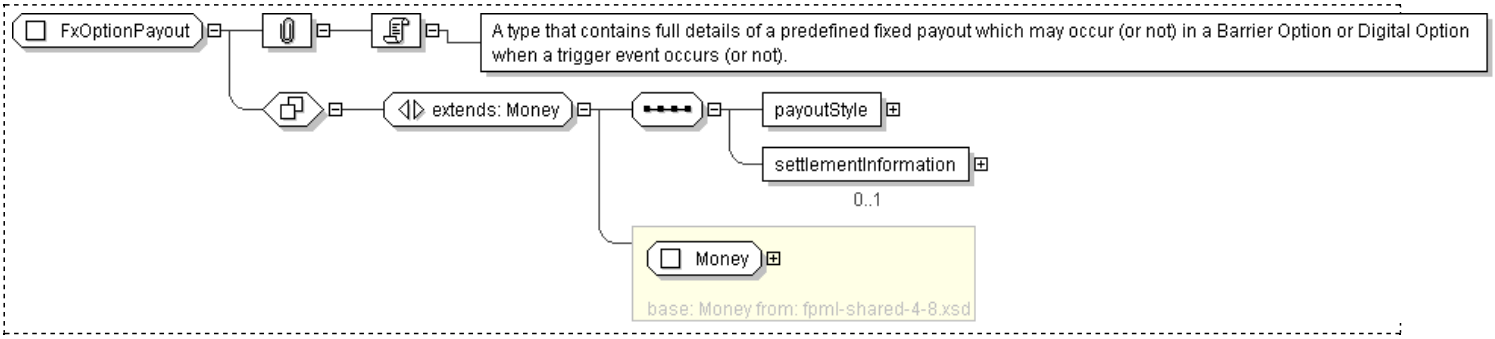
Super-types:	Money < FxOptionPayout (by extension)
Sub-types:	None

Name	FxOptionPayout
Used by (from the same schema document)	Complex Type FxBarrierOption , Complex Type FxDigitalOption
Abstract	no
Documentation	A type that contains full details of a predefined fixed payout which may occur (or not) in a Barrier Option or Digital Option when a trigger event occurs (or not).

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

   <amount> xsd:decimal </amount> [1]
    'The monetary quantity in currency units.'

   <payoutStyle> PayoutEnum </payoutStyle> [1]
    'The trigger event and payout may be asynchonous. A payout may become due on the trigger event, or the
    payout may (by agreeement at initiation) be deferred (for example) to the maturity date.'

   <settlementInformation> SettlementInformation </settlementInformation> [0..1]
    'The information required to settle a currency payment that results from a trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxOptionPayout">
  <xsd:complexContent>
    <xsd:extension base=" Money ">
      <xsd:sequence>
        <xsd:element name="payoutStyle" type=" PayoutEnum "/>
        <xsd:element name="settlementInformation" type=" SettlementInformation " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxOptionPremium

[Table of contents]

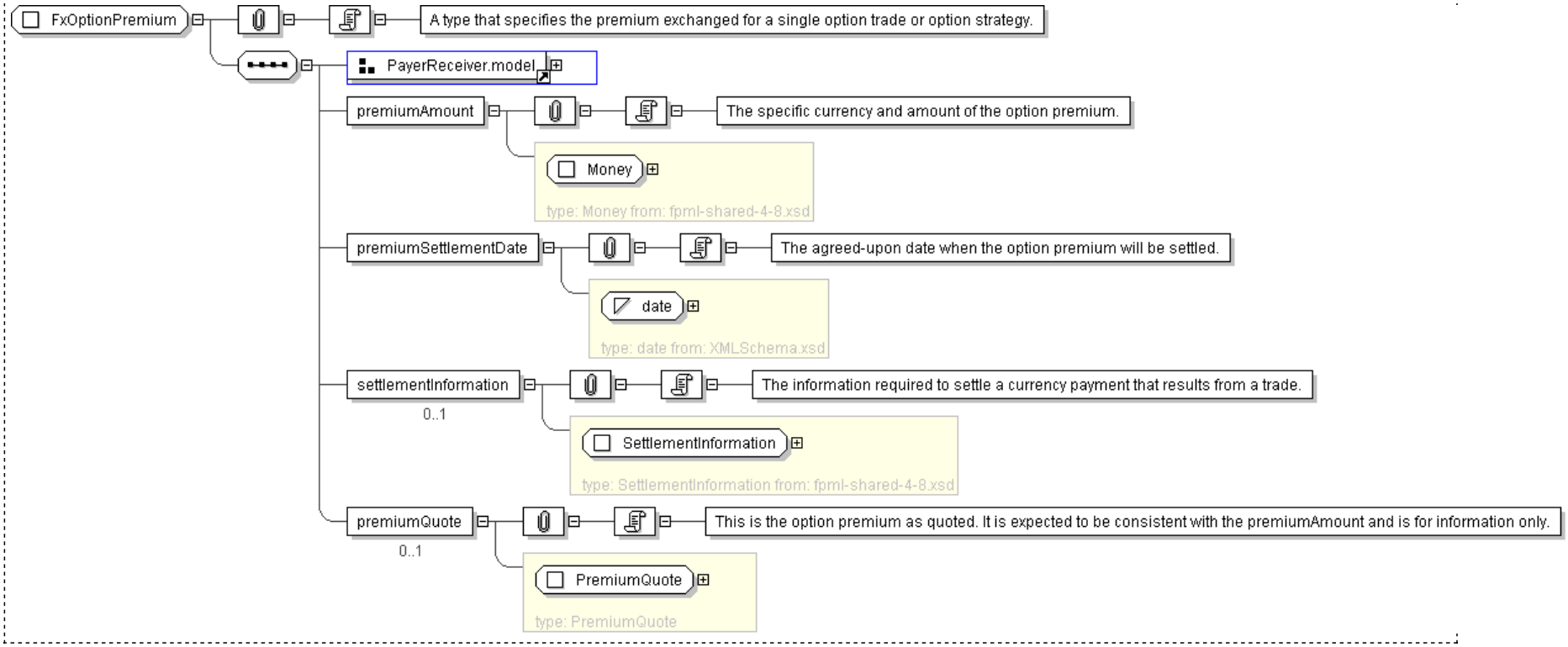
Super-types:	None
Sub-types:	None

Name	FxOptionPremium
Used by (from the same schema document)	Complex Type FxAverageRateOption , Complex Type FxDigitalOption , Complex Type FxOptionLeg
Abstract	no
Documentation	A type that specifies the premium exchanged for a single option trade or option strategy.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <premiumAmount> Money </premiumAmount> [1]
   'The specific currency and amount of the option premium.'

   <premiumSettlementDate> xsd:date </premiumSettlementDate> [1]
   'The agreed-upon date when the option premium will be settled.'

   <settlementInformation> SettlementInformation </settlementInformation> [0..1]
   'The information required to settle a currency payment that results from a trade.'

   <premiumQuote> PremiumQuote </premiumQuote> [0..1]
   'This is the option premium as quoted. It is expected to be consistent with the premiumAmount and is for information only.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxOptionPremium">
  <xsd:sequence>
    <xsd:group ref=" PayerReceiver.model "/>
    <xsd:element name="premiumAmount" type=" Money "/>
    <xsd:element name="premiumSettlementDate" type=" xsd:date "/>
    <xsd:element name="settlementInformation" type=" SettlementInformation " minOccurs="0"/>
    <xsd:element name="premiumQuote" type=" PremiumQuote " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxStrikePrice

[Table of contents]

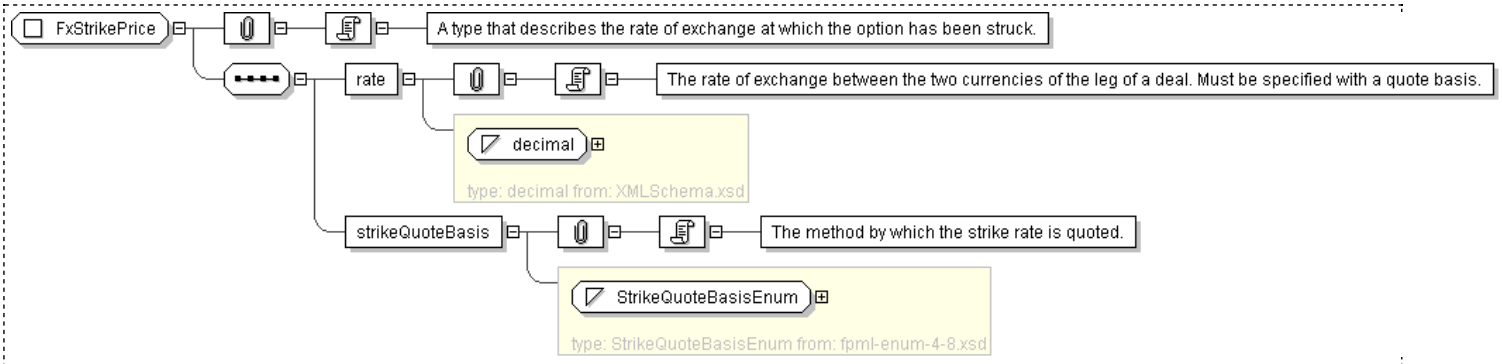
Super-types:	None
Sub-types:	None

Name	FxStrikePrice
Used by (from the same schema document)	Complex Type FxAverageRateOption , Complex Type FxOptionLeg
Abstract	no
Documentation	A type that describes the rate of exchange at which the option has been struck.

XML Instance Representation

```
<...>
  <rate> xsd:decimal </rate> [1]
   'The rate of exchange between the two currencies of the leg of a deal. Must be specified with a quote basis.'

  <strikeQuoteBasis> StrikeQuoteBasisEnum </strikeQuoteBasis> [1]
   'The method by which the strike rate is quoted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxStrikePrice">
  <xsd:sequence>
    <xsd:element name="rate" type=" xsd:decimal "/>
    <xsd:element name="strikeQuoteBasis" type=" StrikeQuoteBasisEnum "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxSwap

[Table of contents]

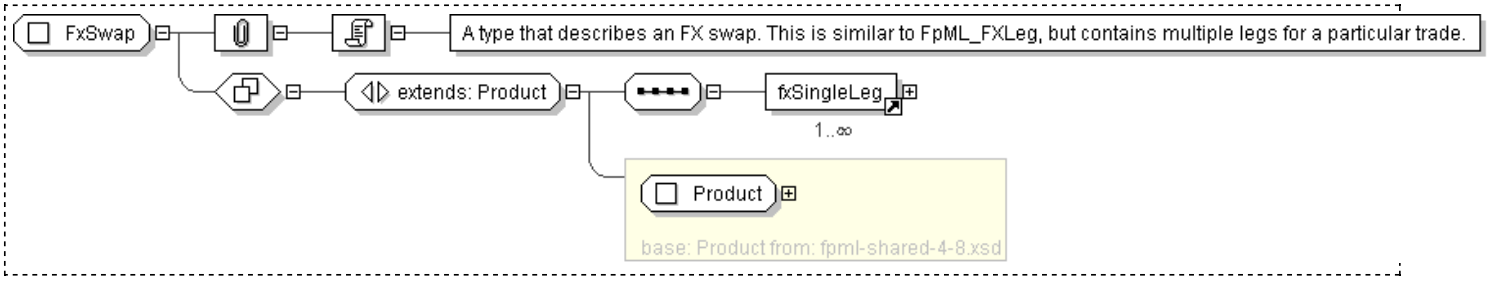
Super-types:	Product < FxSwap (by extension)
Sub-types:	None

Name	FxSwap
Used by (from the same schema document)	Element fxSwap
Abstract	no
Documentation	A type that describes an FX swap. This is similar to FpML_FXLeg, but contains multiple legs for a particular trade.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a
   coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not strictly an
   enumerated list.'

   <fxSingleLeg> ... </fxSingleLeg> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxSwap">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element ref=" fxSingleLeg " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ObservedRates

[Table of contents]

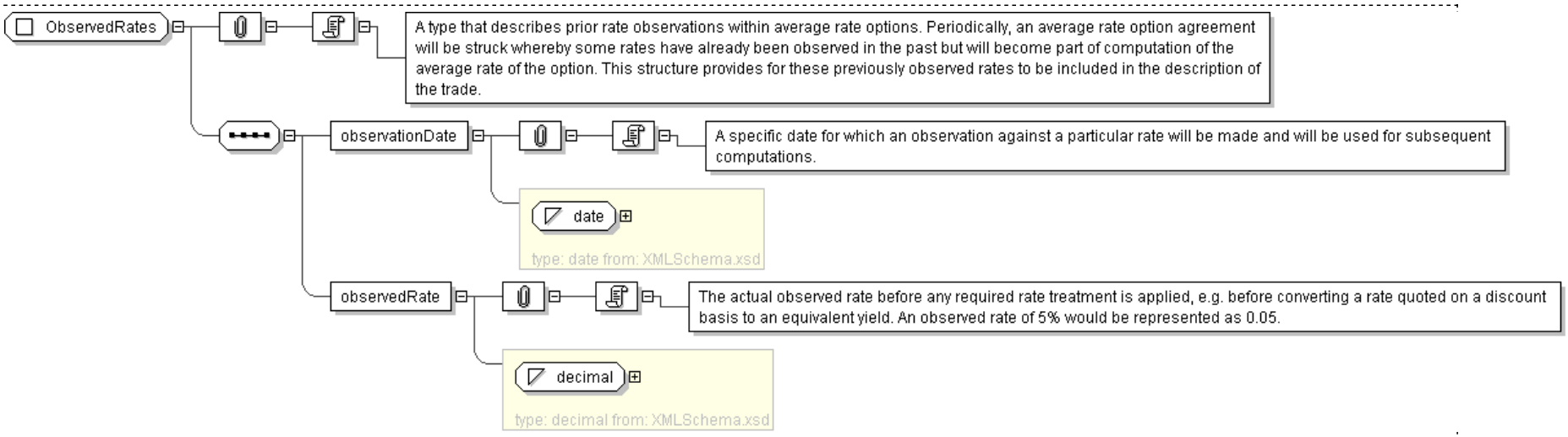
Super-types:	None
Sub-types:	None

Name	ObservedRates
Used by (from the same schema document)	Complex Type FxAverageRateOption
Abstract	no
Documentation	A type that describes prior rate observations within average rate options. Periodically, an average rate option agreement will be struck whereby some rates have already been observed in the past but will become part of computation of the average rate of the option. This structure provides for these previously observed rates to be included in the description of the trade.

XML Instance Representation

```
<...>
   <observationDate> xsd:date </observationDate> [1]
   'A specific date for which an observation against a particular rate will be made and will be used for subsequent computations.'

   <observedRate> xsd:decimal </observedRate> [1]
   'The actual observed rate before any required rate treatment is applied, e.g. before converting a rate quoted on a discount basis to
   an equivalent yield. An observed rate of 5% would be represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ObservedRates">
   <xsd:sequence>
      <xsd:element name="observationDate" type=" xsd:date "/>
      <xsd:element name="observedRate" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PremiumQuote

[Table of contents]

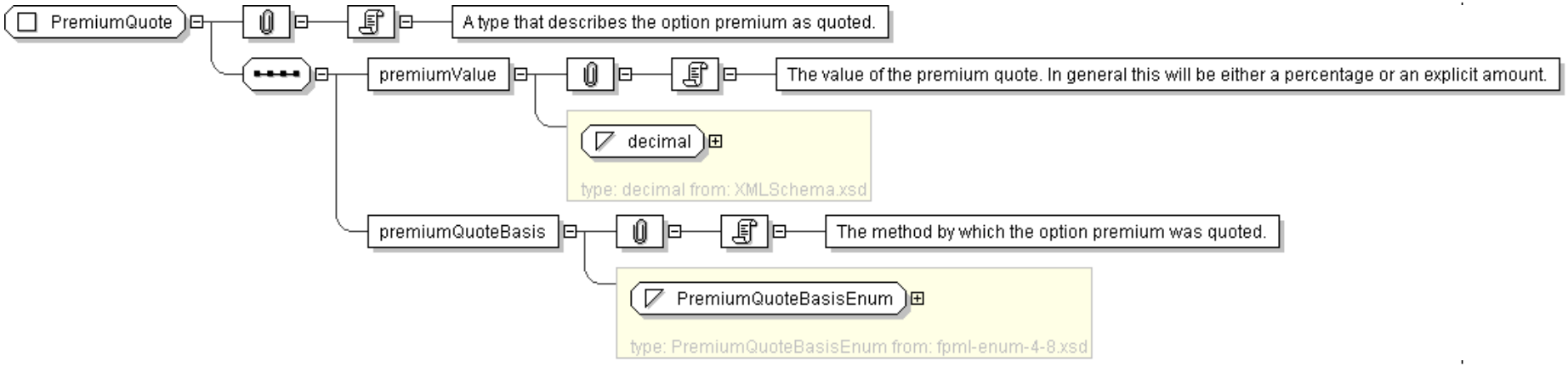
Super-types:	None
Sub-types:	None

Name	PremiumQuote
Used by (from the same schema document)	Complex Type FxOptionPremium
Abstract	no
Documentation	A type that describes the option premium as quoted.

XML Instance Representation

```
<...>
   <premiumValue> xsd:decimal </premiumValue> [1]
   'The value of the premium quote. In general this will be either a percentage or an explicit amount.'

   <premiumQuoteBasis> PremiumQuoteBasisEnum </premiumQuoteBasis> [1]
   'The method by which the option premium was quoted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PremiumQuote">
   <xsd:sequence>
      <xsd:element name="premiumValue" type=" xsd:decimal "/>
      <xsd:element name="premiumQuoteBasis" type=" PremiumQuoteBasisEnum "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotedAs

[Table of contents]

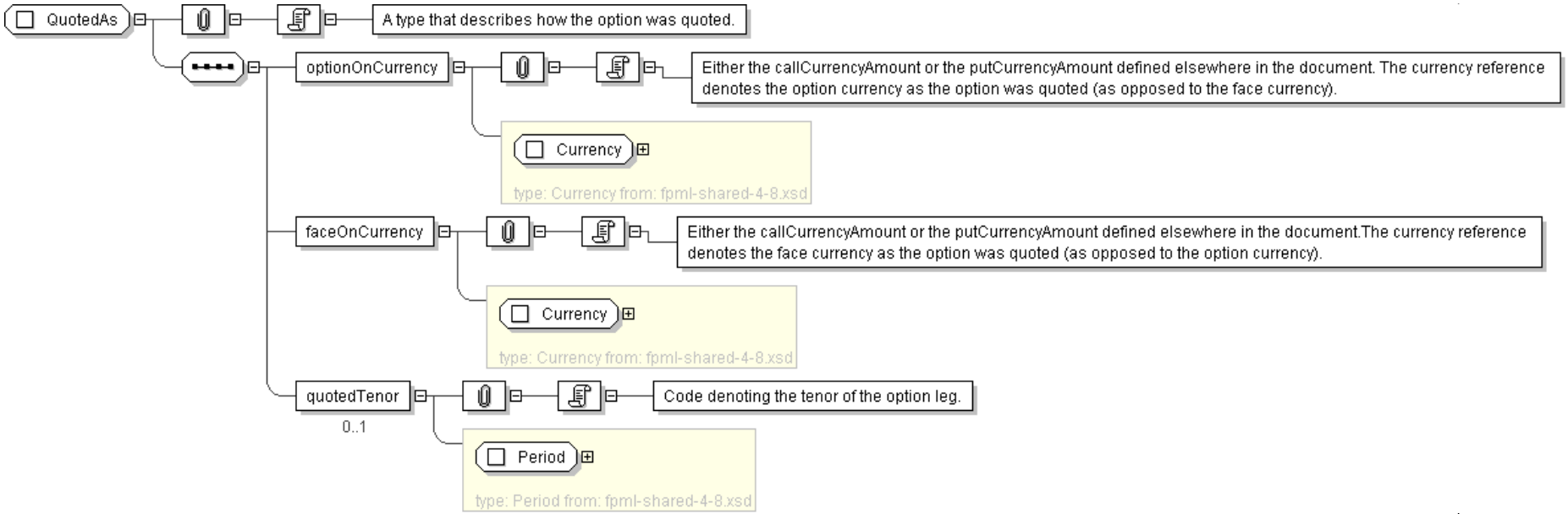
Super-types:	None
Sub-types:	None

Name	QuotedAs
Used by (from the same schema document)	Complex Type FxOptionLeg
Abstract	no
Documentation	A type that describes how the option was quoted.

XML Instance Representation

```
<...>
   <optionOnCurrency> Currency </optionOnCurrency> [1]
    'Either the callCurrencyAmount or the putCurrencyAmount defined elsewhere in the document. The currency reference denotes the
    option currency as the option was quoted (as opposed to the face currency).'

   <faceOnCurrency> Currency </faceOnCurrency> [1]
    'Either the callCurrencyAmount or the putCurrencyAmount defined elsewhere in the document.The currency reference denotes the face
    currency as the option was quoted (as opposed to the option currency).'

   <quotedTenor> Period </quotedTenor> [0..1]
    'Code denoting the tenor of the option leg.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotedAs">
  <xsd:sequence>
    <xsd:element name="optionOnCurrency" type=" Currency "/>
    <xsd:element name="faceOnCurrency" type=" Currency "/>
    <xsd:element name="quotedTenor" type=" Period " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SideRate

[Table of contents]

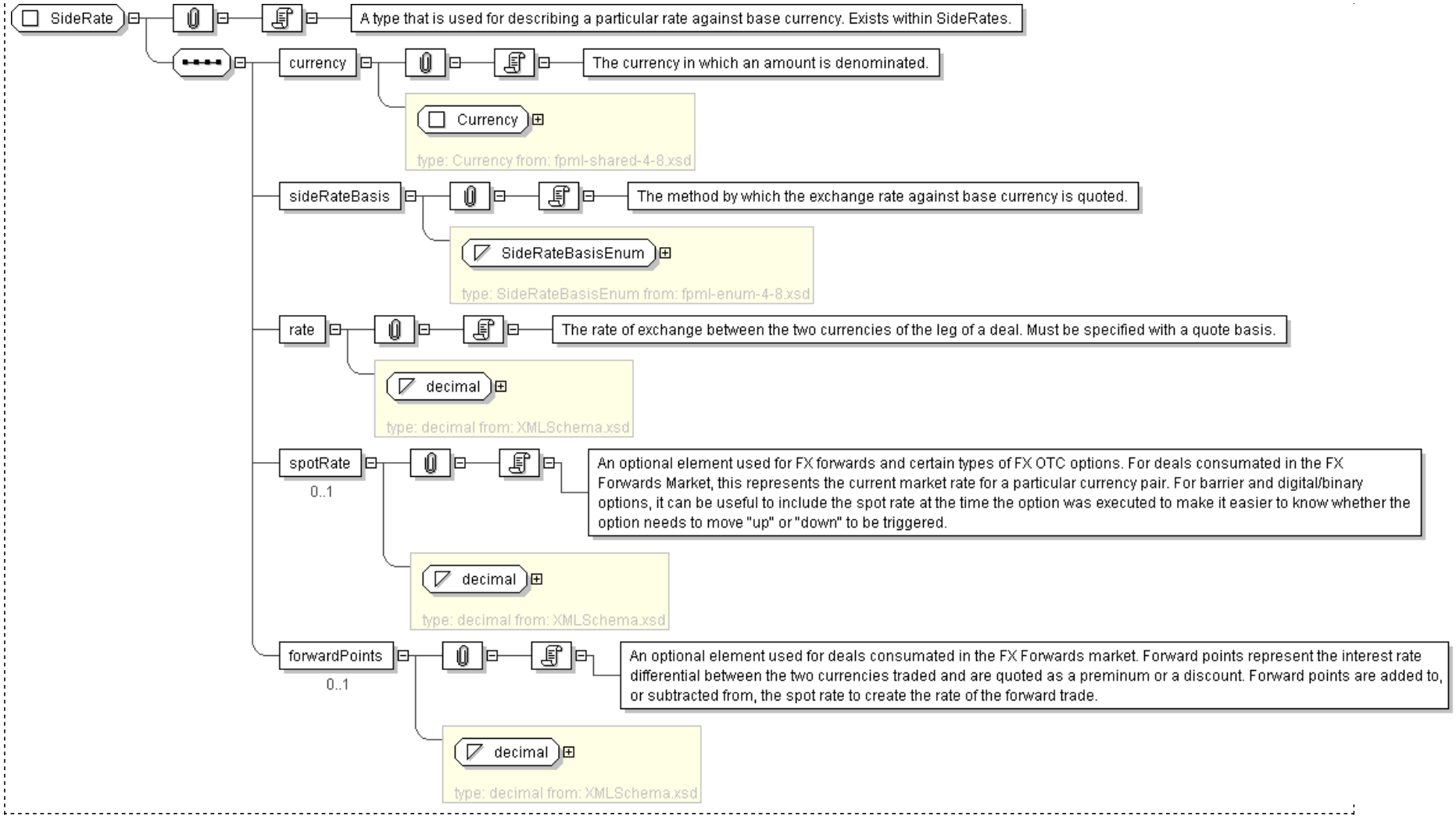
Super-types:	None
Sub-types:	None

Name	SideRate
Used by (from the same schema document)	Complex Type SideRates , Complex Type SideRates
Abstract	no
Documentation	A type that is used for describing a particular rate against base currency. Exists within SideRates.

XML Instance Representation

```
<...>
   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

   <sideRateBasis> SideRateBasisEnum </sideRateBasis> [1]
   'The method by which the exchange rate against base currency is quoted.'

   <rate> xsd:decimal </rate> [1]
   'The rate of exchange between the two currencies of the leg of a deal. Must be specified with a quote basis.'

   <spotRate> xsd:decimal </spotRate> [0..1]
   'An optional element used for FX forwards and certain types of FX OTC options. For deals consumated in the FX Forwards Market,
   this represents the current market rate for a particular currency pair. For barrier and digital/binary options, it can be useful
   to include the spot rate at the time the option was executed to make it easier to know whether the option needs to move \"up\" or
   \"down\" to be triggered.'

   <forwardPoints> xsd:decimal </forwardPoints> [0..1]
   'An optional element used for deals consumated in the FX Forwards market. Forward points represent the interest rate differential
   between the two currencies traded and are quoted as a preminum or a discount. Forward points are added to, or subtracted from,
   the spot rate to create the rate of the forward trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SideRate">
  <xsd:sequence>
    <xsd:element name="currency" type=" Currency "/>
    <xsd:element name="sideRateBasis" type=" SideRateBasisEnum "/>
    <xsd:element name="rate" type=" xsd:decimal "/>
    <xsd:element name="spotRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="forwardPoints" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SideRates

[Table of contents]

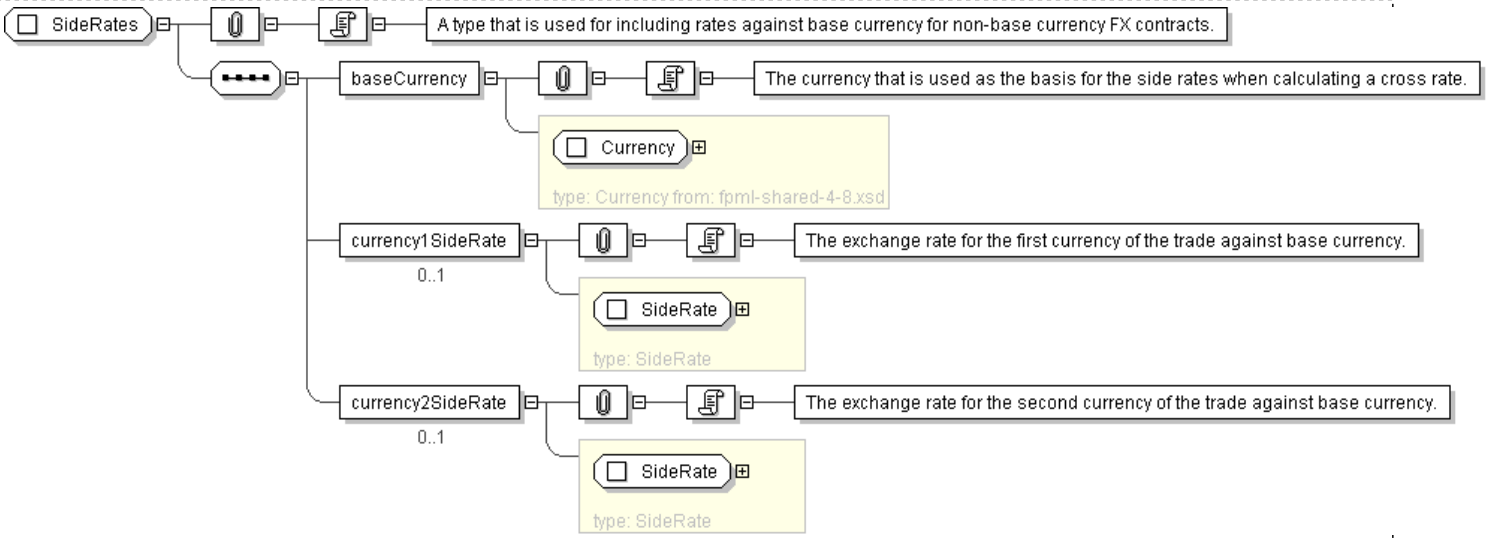
Super-types:	None
Sub-types:	None

Name	SideRates
Used by (from the same schema document)	Complex Type ExchangeRate
Abstract	no
Documentation	A type that is used for including rates against base currency for non-base currency FX contracts.

XML Instance Representation

```
<...>
  <baseCurrency> Currency </baseCurrency> [1]
  'The currency that is used as the basis for the side rates when calculating a cross rate.'

  <currency1SideRate> SideRate </currency1SideRate> [0..1]
  'The exchange rate for the first currency of the trade against base currency.'

  <currency2SideRate> SideRate </currency2SideRate> [0..1]
  'The exchange rate for the second currency of the trade against base currency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SideRates">
  <xsd:sequence>
    <xsd:element name="baseCurrency" type=" Currency "/>
    <xsd:element name="currency1SideRate" type=" SideRate " minOccurs="0"/>
    <xsd:element name="currency2SideRate" type=" SideRate " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TermDeposit

[Table of contents]

Super-types:	Product < TermDeposit (by extension)
Sub-types:	None

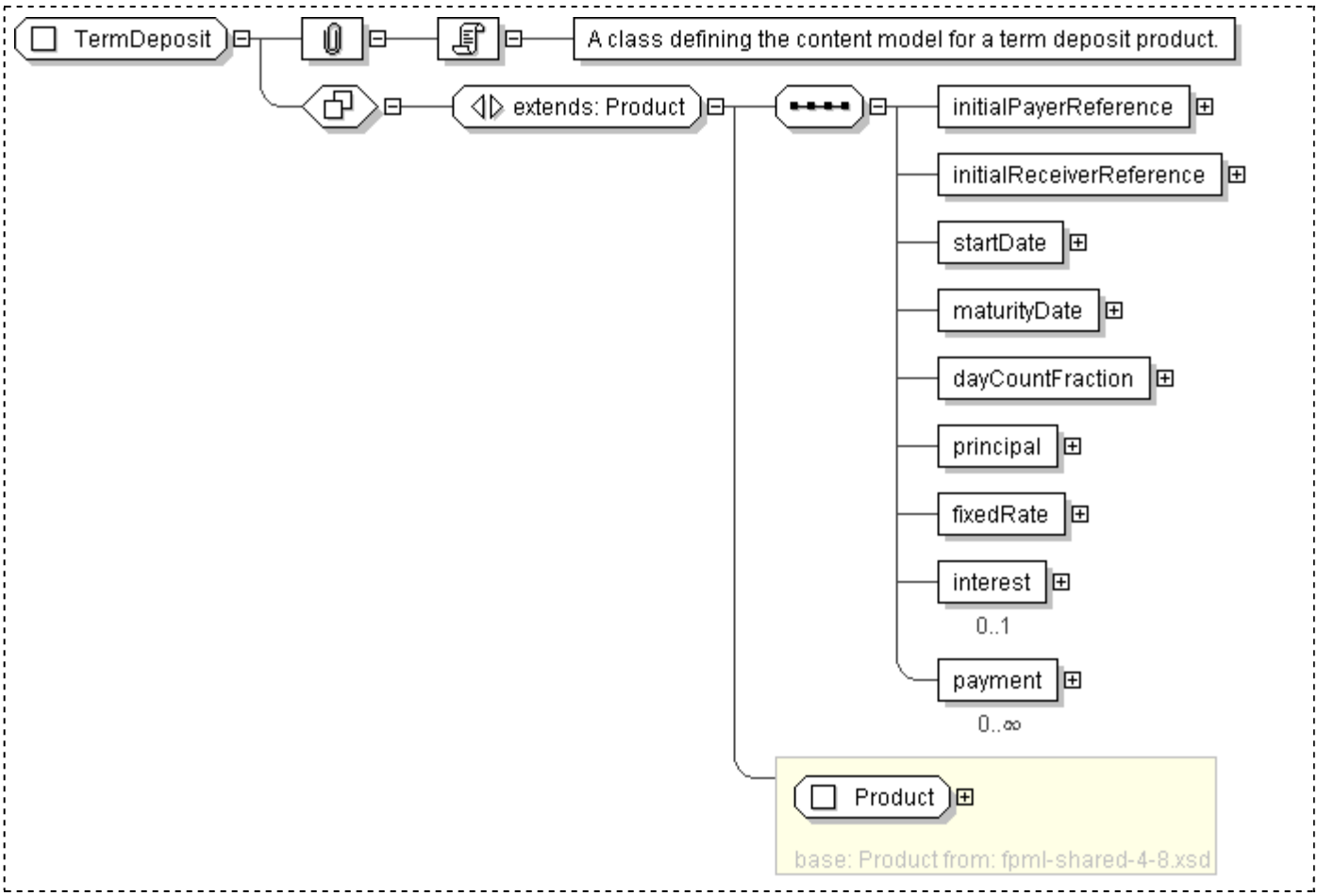
Name	TermDeposit
Used by (from the same schema document)	Element termDeposit
Abstract	no
Documentation	A class defining the content model for a term deposit product.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <initialPayerReference> PartyReference </initialPayerReference> [1]
    'A pointer style reference to a party identifier defined elsewhere in the
    document. The party referenced is the payer of the initial principal of the
    deposit on the start date.'

   <initialReceiverReference> PartyReference </initialReceiverReference> [1]
    'A pointer style reference to a party identifier defined elsewhere in the
    document. The party is the receiver of the initial principal of the deposit on
    the start date.'

   <startDate> xsd:date </startDate> [1]
    'The averaging period start date.'

   <maturityDate> xsd:date </maturityDate> [1]
    'The end date of the calculation period. This date should already be adjusted
    for any applicable business day convention.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [1]
    'The day count fraction.'

   <principal> Money </principal> [1]
    'The principal amount of the trade.'

   <fixedRate> xsd:decimal </fixedRate> [1]
    'The calculation period fixed rate. A per annum rate, expressed as a decimal. A
    fixed rate of 5% would be represented as 0.05.'

   <interest> Money </interest> [0..1]
    'The total interest of at maturity of the trade.'

   <payment> Payment </payment> [0..*]
    'A known payment between two parties.'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TermDeposit">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="initialPayerReference" type=" PartyReference "/>
            <xsd:element name="initialReceiverReference" type=" PartyReference "/>
            <xsd:element name="startDate" type=" xsd:date "/>
            <xsd:element name="maturityDate" type=" xsd:date "/>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "/>
            <xsd:element name="principal" type=" Money "/>
            <xsd:element name="fixedRate" type=" xsd:decimal "/>
            <xsd:element name="interest" type=" Money " minOccurs="0"/>
            <xsd:element name="payment" type=" Payment " minOccurs="0"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: bulletPayment
  - Element: capFloor
  - Element: floatingRateCalculation
  - Element: fra
  - Element: inflationRateCalculation
  - Element: rateCalculation
  - Element: swap
  - Element: swaption
- Global Definitions
  - Complex Type: BondReference
  - Complex Type: BulletPayment
  - Complex Type: Calculation
  - Complex Type: CalculationPeriod
  - Complex Type: CalculationPeriodAmount
  - Complex Type: CalculationPeriodDates
  - Complex Type: CalculationPeriodDatesReference
  - Complex Type: CancelableProvision
  - Complex Type: CancelableProvisionAdjustedDates
  - Complex Type: CancellationEvent
  - Complex Type: CapFloor
  - Complex Type: CashPriceMethod
  - Complex Type: CashSettlement
  - Complex Type: CashSettlementPaymentDate
  - Complex Type: Cashflows
  - Complex Type: DateRelativeToCalculationPeriodDates
  - Complex Type: DateRelativeToPaymentDates
  - Complex Type: Discounting
  - Complex Type: EarlyTerminationEvent
  - Complex Type: EarlyTerminationProvision
  - Complex Type: ExerciseEvent
  - Complex Type: ExercisePeriod
  - Complex Type: ExtendibleProvision
  - Complex Type: ExtendibleProvisionAdjustedDates
  - Complex Type: ExtensionEvent
  - Complex Type: FallbackReferencePrice
  - Complex Type: FinalCalculationPeriodDateAdjustment
  - Complex Type: FloatingRateDefinition
  - Complex Type: Fra
  - Complex Type: FxFixingDate
  - Complex Type: FxLinkedNotionalAmount
  - Complex Type: FxLinkedNotionalSchedule
  - Complex Type: InflationRateCalculation
  - Complex Type: InterestRateStream
  - Complex Type: InterestRateStreamReference
  - Complex Type: MandatoryEarlyTermination
  - Complex Type: MandatoryEarlyTerminationAdjustedDates
  - Complex Type: NonDeliverableSettlement
  - Complex Type: Notional
  - Complex Type: NotionalStepRule
  - Complex Type: OptionalEarlyTermination
  - Complex Type: OptionalEarlyTerminationAdjustedDates
  - Complex Type: PaymentCalculationPeriod

- Complex Type: PaymentDates
- Complex Type: PaymentDatesReference
- Complex Type: PriceSourceDisruption
- Complex Type: PrincipalExchange
- Complex Type: RelevantUnderlyingDateReference
- Complex Type: ResetDates
- Complex Type: ResetDatesReference
- Complex Type: SettlementProvision
- Complex Type: SettlementRateOption
- Complex Type: SinglePartyOption
- Complex Type: StubCalculationPeriodAmount
- Complex Type: Swap
- Complex Type: SwapAdditionalTerms
- Complex Type: Swaption
- Complex Type: SwaptionAdjustedDates
- Complex Type: ValuationDatesReference
- Complex Type: ValuationPostponement
- Complex Type: YieldCurveMethod
- Model Group: DiscountRate.model
- Model Group: MandatoryEarlyTermination.model
- Model Group: OptionalEarlyTermination.model

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ○ fpml-asset-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-asset-4-8.xsd"/>
  ...
```

```
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
```

```
          </element>
        </sequence>
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

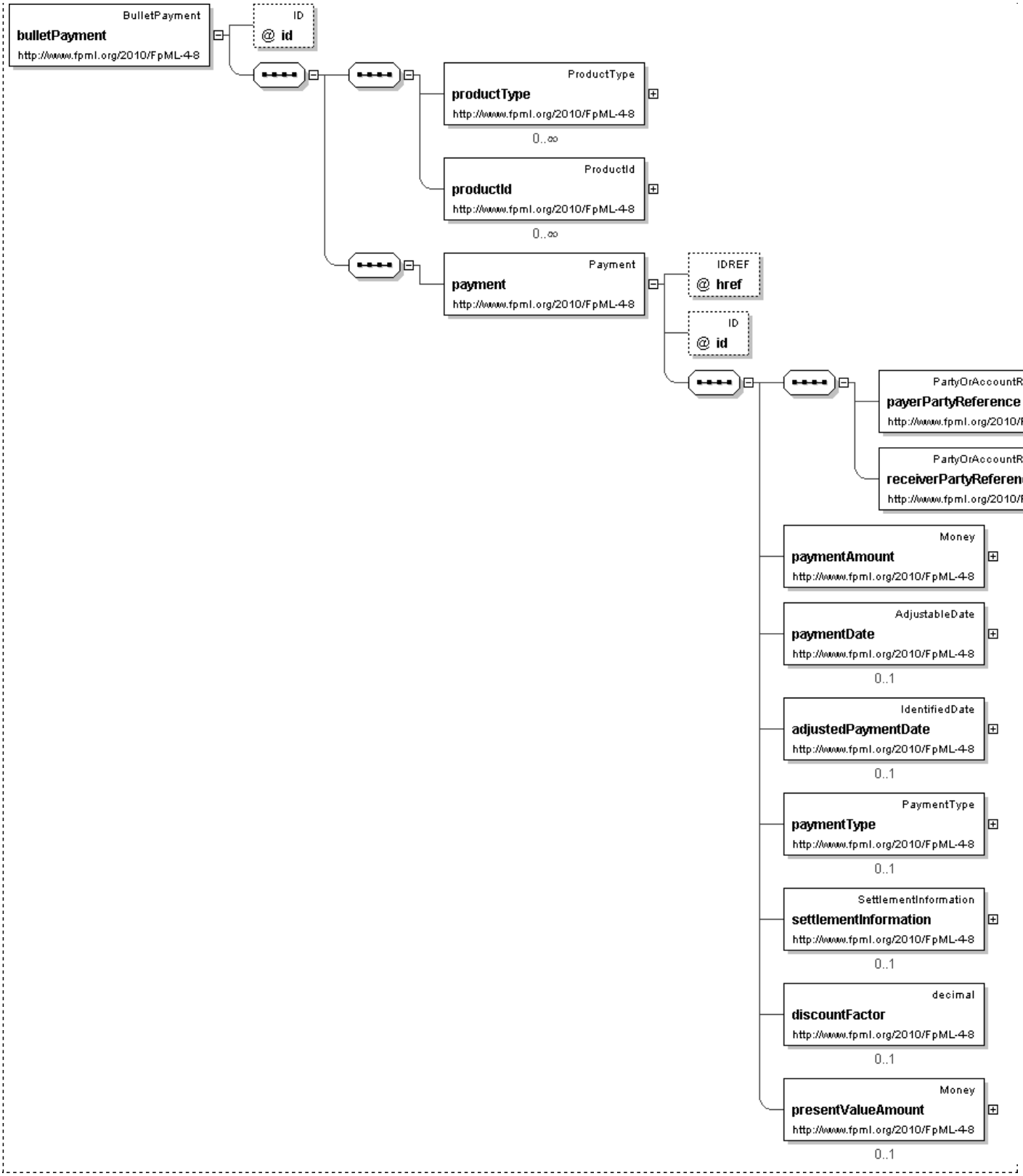
Element: bulletPayment

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	bulletPayment
Type	BulletPayment
Nillable	no
Abstract	no
Documentation	A product to represent a single known payment.

Logical Diagram

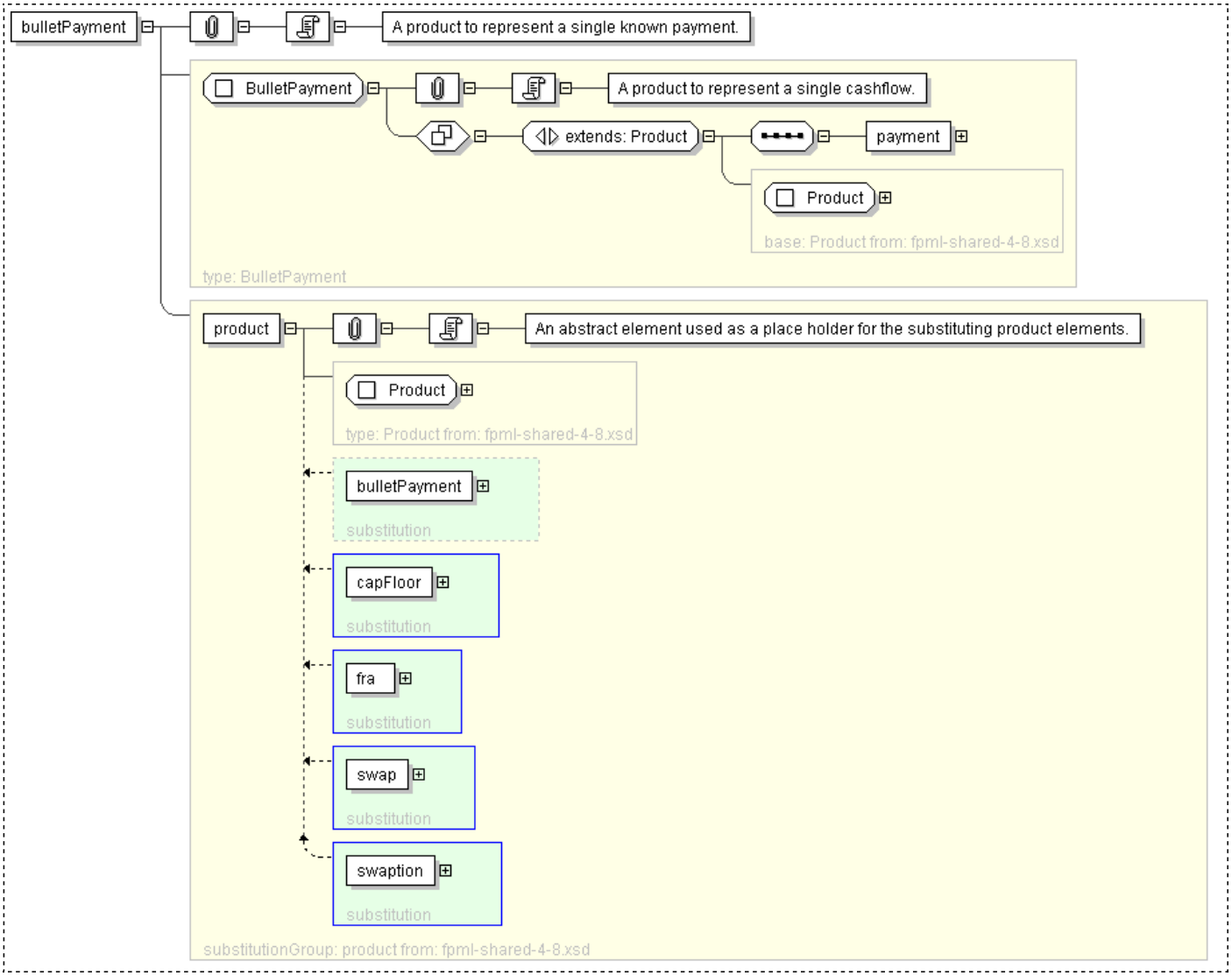


XML Instance Representation

```
<bulletPayment
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated
   with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <payment> Payment </payment> [1]
   'A known payment between two parties.'
```

```
</bulletPayment>
```

Diagram



Schema Component Representation

```
<xsd:element name="bulletPayment" type=" BulletPayment " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

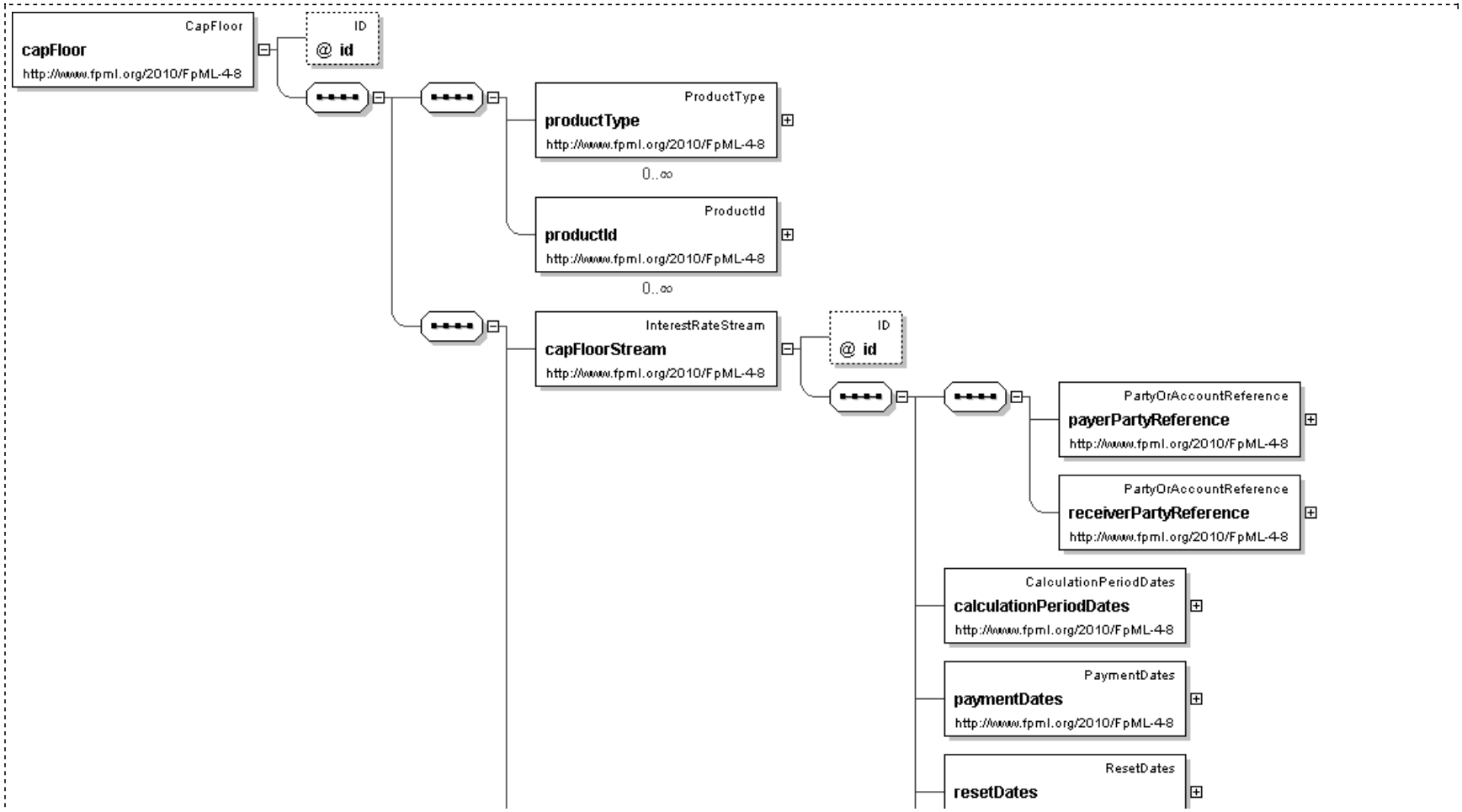
Element: capFloor

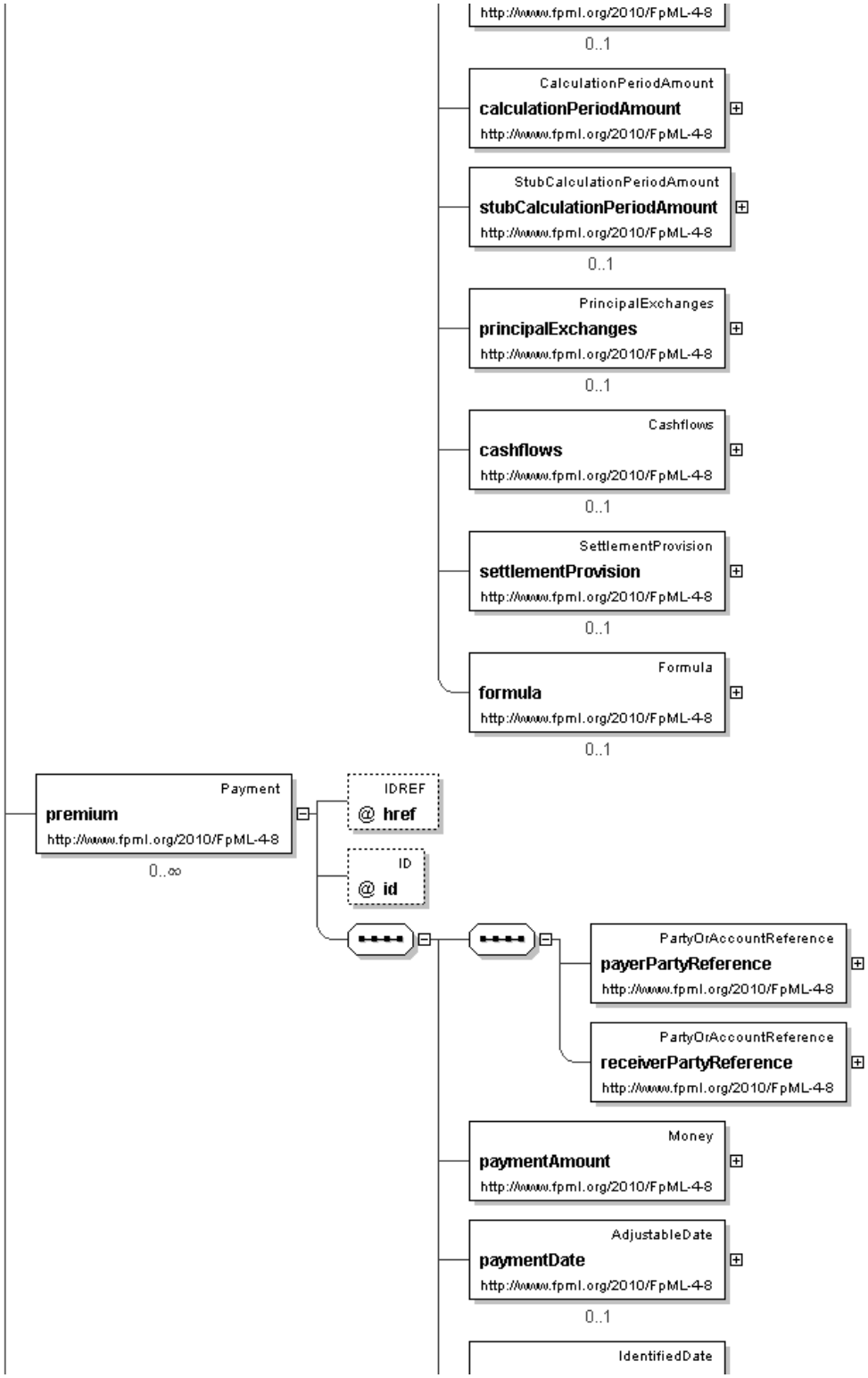
[Table of contents]

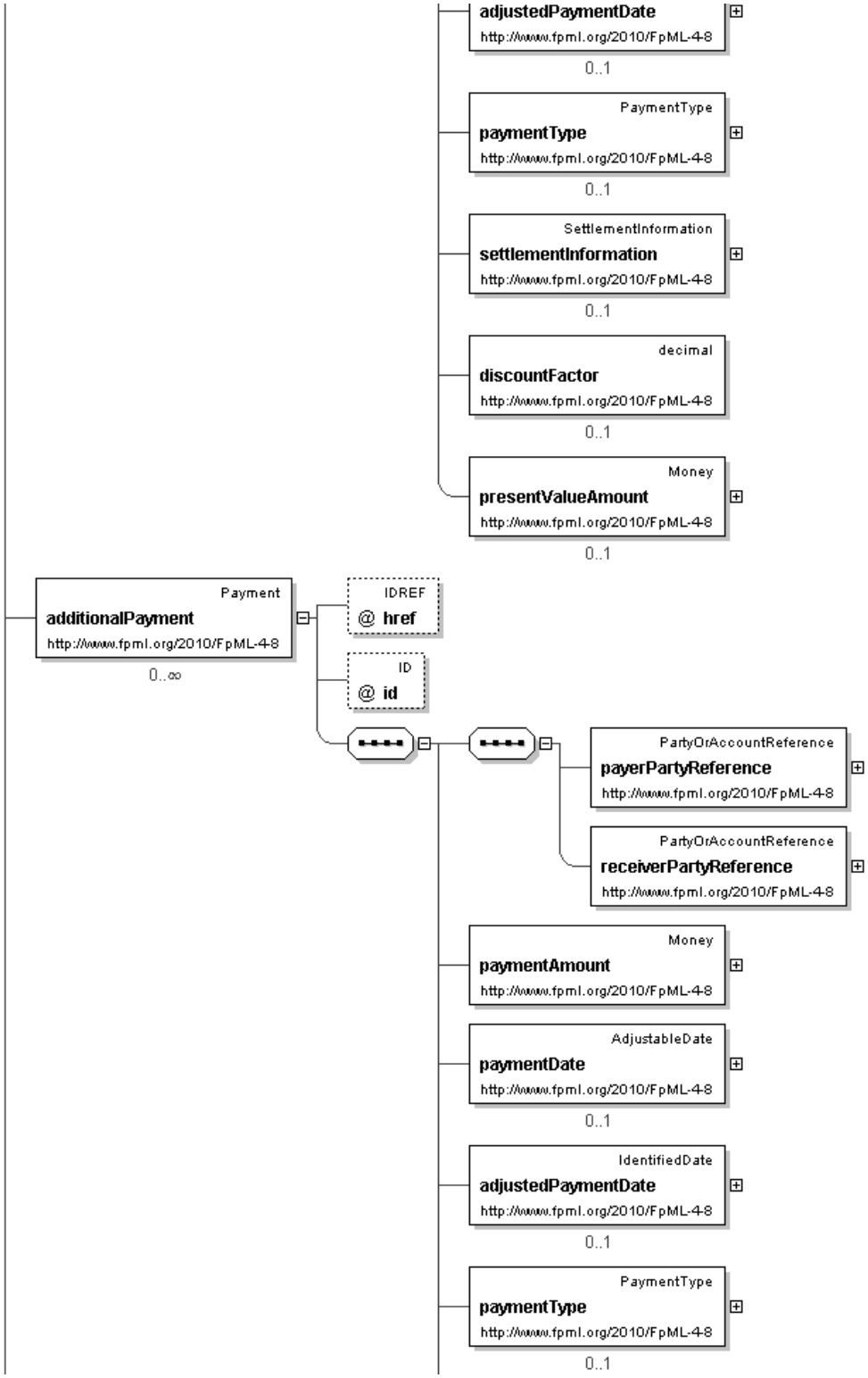
- *This element can be used wherever the following element is referenced:*
  - product

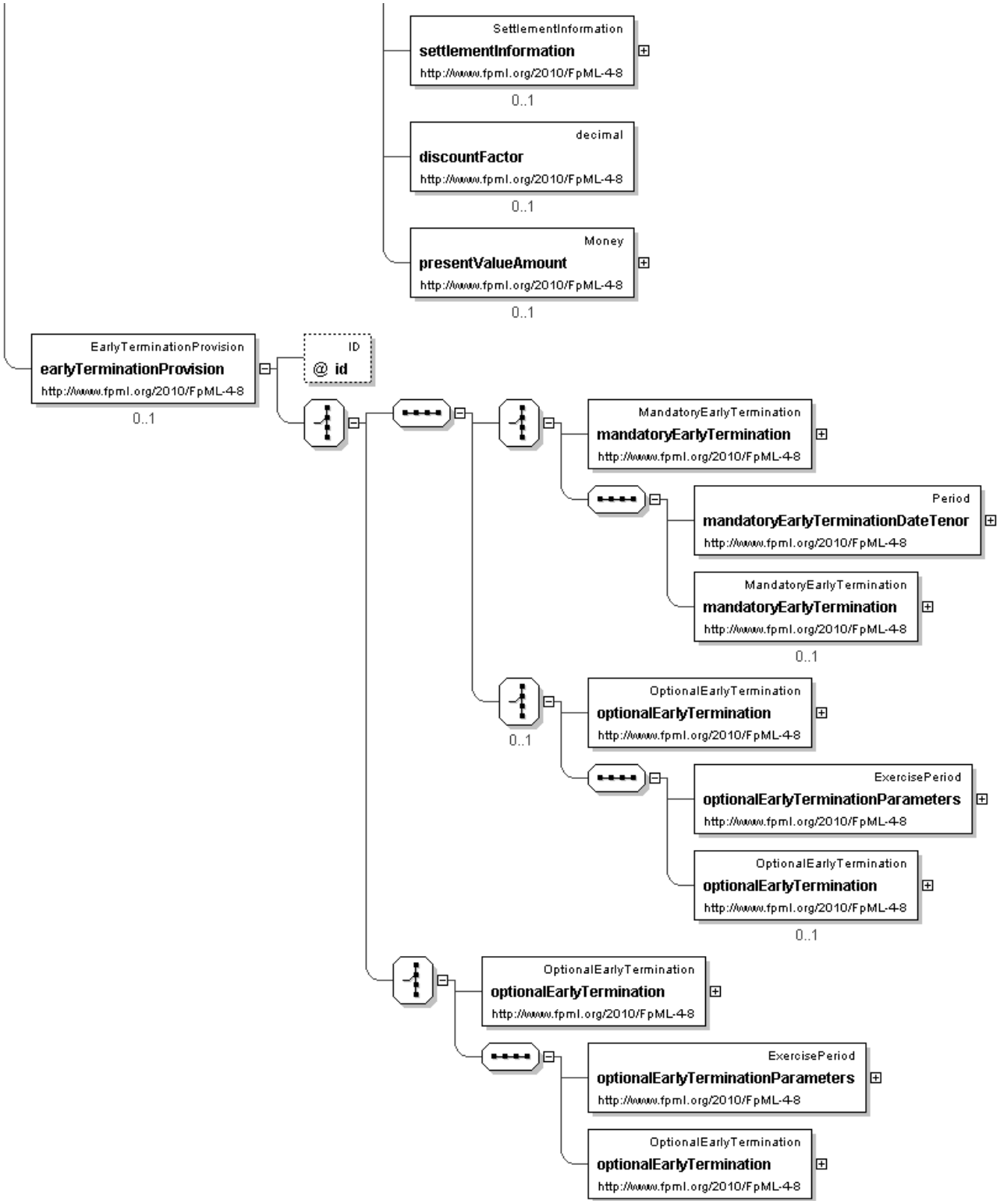
Name	capFloor
Type	CapFloor
Nillable	no
Abstract	no
Documentation	A cap, floor or cap floor structures product definition.

Logical Diagram




http://www.fpml.org/2010/FpML-4-8
0..1
CalculationPeriodAmount
calculationPeriodAmount
http://www.fpml.org/2010/FpML-4-8
StubCalculationPeriodAmount
stubCalculationPeriodAmount
http://www.fpml.org/2010/FpML-4-8
0..1
PrincipalExchanges
principalExchanges
http://www.fpml.org/2010/FpML-4-8
0..1
Cashflows
cashflows
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementProvision
settlementProvision
http://www.fpml.org/2010/FpML-4-8
0..1
Formula
formula
http://www.fpml.org/2010/FpML-4-8
0..1
Payment
premium
http://www.fpml.org/2010/FpML-4-8
0..∞
IDREF
@ href
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate


adjustedPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PaymentType
paymentType
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Payment
additionalPayment
http://www.fpml.org/2010/FpML-4-8
0..∞
IDREF
@ href
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
adjustedPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PaymentType
paymentType
http://www.fpml.org/2010/FpML-4-8
0..1

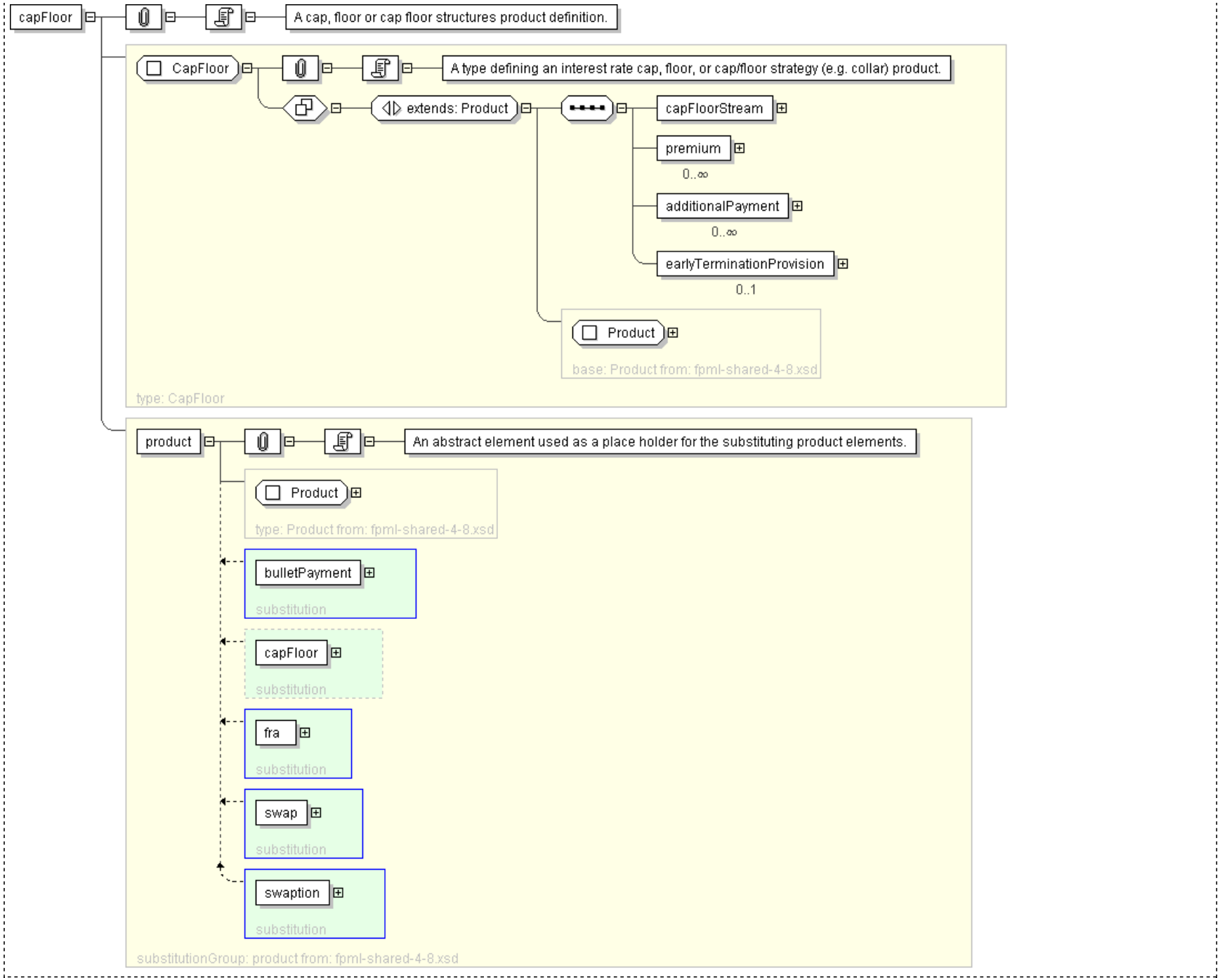

SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
EarlyTerminationProvision
earlyTerminationProvision
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
MandatoryEarlyTermination
mandatoryEarlyTermination
http://www.fpml.org/2010/FpML-4-8
Period
mandatoryEarlyTerminationDateTenor
http://www.fpml.org/2010/FpML-4-8
MandatoryEarlyTermination
mandatoryEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
ExercisePeriod
optionalEarlyTerminationParameters
http://www.fpml.org/2010/FpML-4-8
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
ExercisePeriod
optionalEarlyTerminationParameters
http://www.fpml.org/2010/FpML-4-8
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8

0..1

XML Instance Representation

```
<capFloor
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
   that the domain values for this element are not strictly an enumerated list.'

   <capFloorStream> InterestRateStream </capFloorStream> [1]
   <premium> Payment </premium> [0..*]
   'The option premium amount payable by buyer to seller on the specified payment date.'

   <additionalPayment> Payment </additionalPayment> [0..*]
   'Additional payments between the principal parties.'

   <earlyTerminationProvision> EarlyTerminationProvision </earlyTerminationProvision> [0..1]
   'Parameters specifying provisions relating to the optional and mandatory early terminarion of a CapFloor transaction.'

</capFloor>
```

Diagram



Schema Component Representation

```
<xsd:element name="capFloor" type=" CapFloor " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

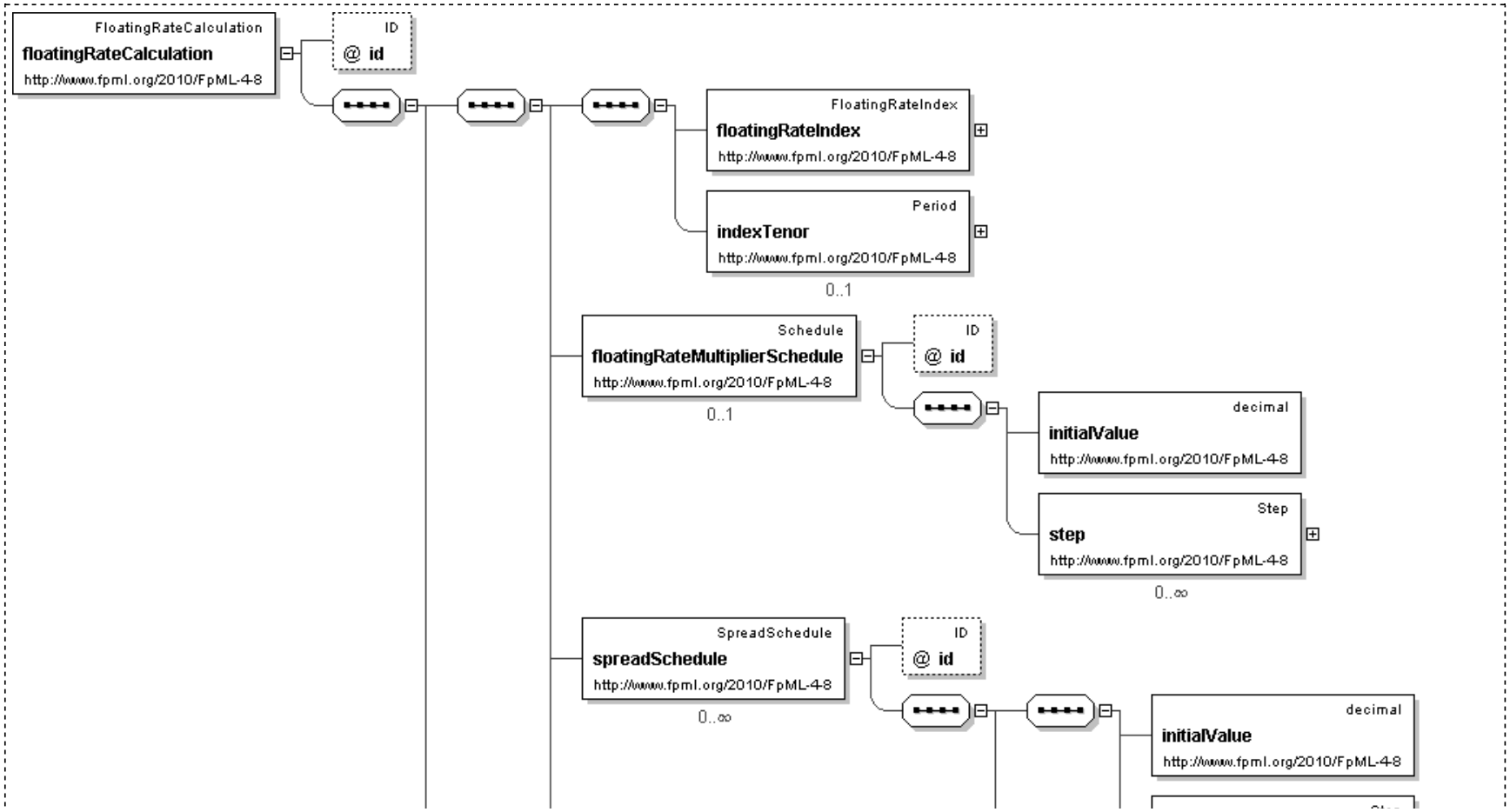
Element: floatingRateCalculation

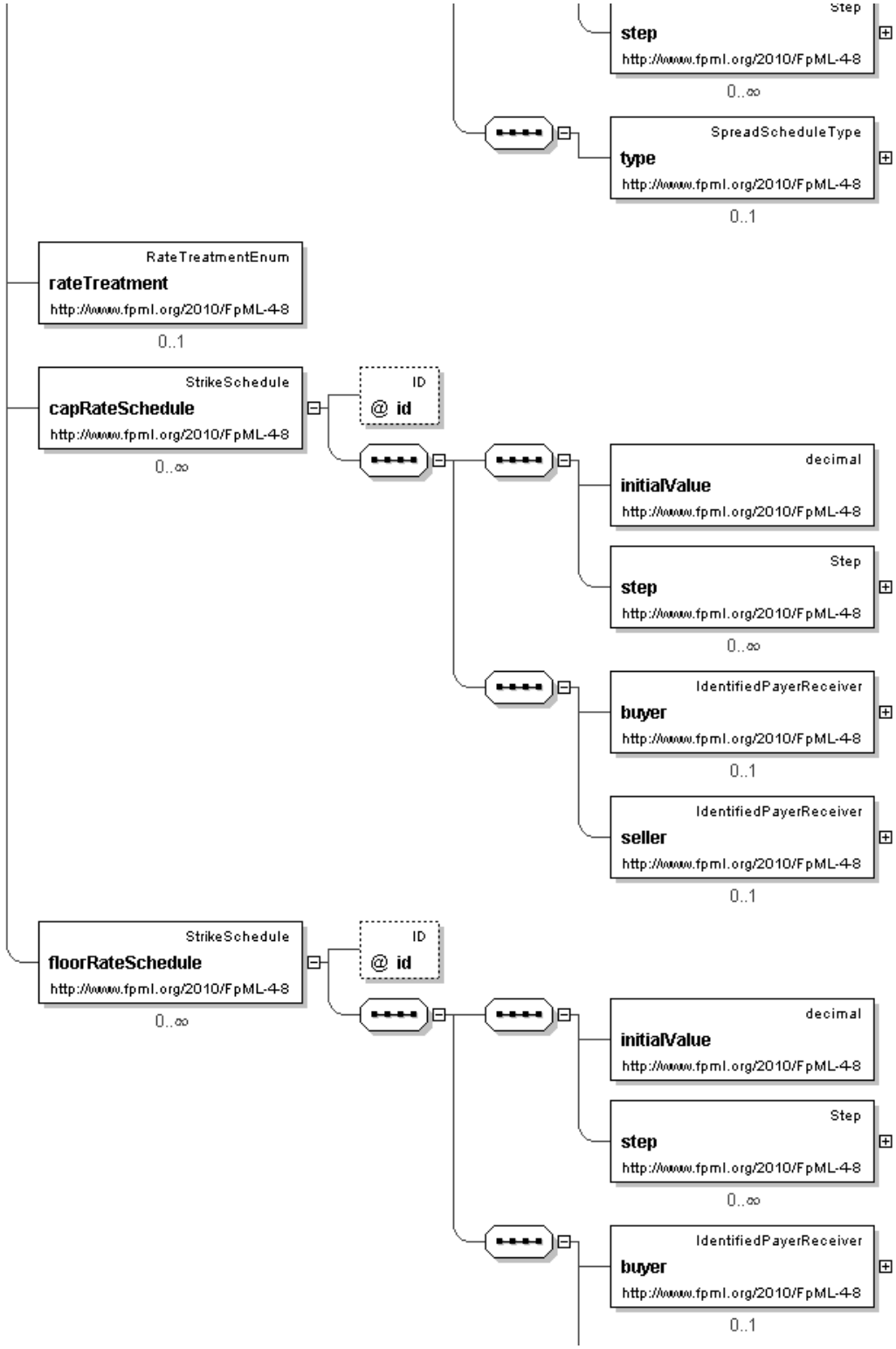
[Table of contents]

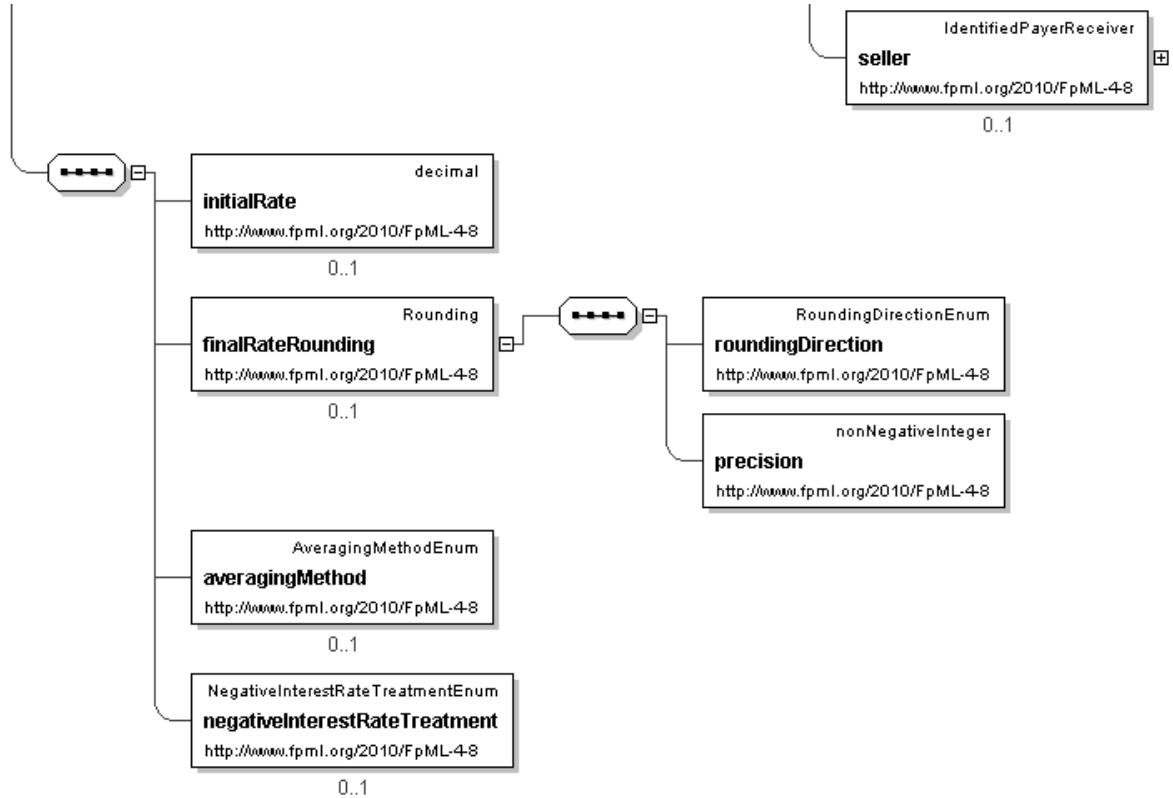
- *This element can be used wherever the following element is referenced:*
  - rateCalculation

Name	floatingRateCalculation
Type	FloatingRateCalculation
Nillable	no
Abstract	no
Documentation	A floating rate calculation definition.

Logical Diagram




Step
step
http://www.fpml.org/2010/FpML-4-8
0..∞
SpreadScheduleType
type
http://www.fpml.org/2010/FpML-4-8
0..1
RateTreatmentEnum
rateTreatment
http://www.fpml.org/2010/FpML-4-8
0..1
StrikeSchedule
capRateSchedule
http://www.fpml.org/2010/FpML-4-8
0..∞
ID
@ id
decimal
initialValue
http://www.fpml.org/2010/FpML-4-8
Step
step
http://www.fpml.org/2010/FpML-4-8
0..∞
IdentifiedPayerReceiver
buyer
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedPayerReceiver
seller
http://www.fpml.org/2010/FpML-4-8
0..1
StrikeSchedule
floorRateSchedule
http://www.fpml.org/2010/FpML-4-8
0..∞
ID
@ id
decimal
initialValue
http://www.fpml.org/2010/FpML-4-8
Step
step
http://www.fpml.org/2010/FpML-4-8
0..∞
IdentifiedPayerReceiver
buyer
http://www.fpml.org/2010/FpML-4-8
0..1



XML Instance Representation

```
<floatingRateCalculation
 id=" xsd:ID [0..1]">
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <floatingRateMultiplierSchedule> Schedule </floatingRateMultiplierSchedule> [0..1]
   'A rate multiplier or multiplier schedule to apply to the floating rate. A multiplier schedule is expressed as explicit
   multipliers and dates. In the case of a schedule, the step dates may be subject to adjustment in accordance with any
   adjustments specified in the calculationPeriodDatesAdjustments. The multiplier can be a positive or negative decimal. This
   element should only be included if the multiplier is not equal to 1 (one) for the term of the stream.'

   <spreadSchedule> SpreadSchedule </spreadSchedule> [0..*]
   'The ISDA Spread or a Spread schedule expressed as explicit spreads and dates. In the case of a schedule, the step dates
   may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments. The spread
   is a per annum rate, expressed as a decimal. For purposes of determining a calculation period amount, if positive the
   spread will be added to the floating rate and if negative the spread will be subtracted from the floating rate. A positive
   10 basis point (0.1%) spread would be represented as 0.001.'

   <rateTreatment> RateTreatmentEnum </rateTreatment> [0..1]
   'The specification of any rate conversion which needs to be applied to the observed rate before being used in any
   calculations. The two common conversions are for securities quoted on a bank discount basis which will need to be
   converted to either a Money Market Yield or Bond Equivalent Yield. See the Annex to the 2000 ISDA Definitions, Section
   7.3. Certain General Definitions Relating to Floating Rate Options, paragraphs (g) and (h) for definitions of these terms.'

   <capRateSchedule> StrikeSchedule </capRateSchedule> [0..*]
   'The cap rate or cap rate schedule, if any, which applies to the floating rate. The cap rate (strike) is only required
   where the floating rate on a swap stream is capped at a certain level. A cap rate schedule is expressed as explicit cap
   rates and dates and the step dates may be subject to adjustment in accordance with any adjustments specified in
   calculationPeriodDatesAdjustments. The cap rate is assumed to be exclusive of any spread and is a per annum rate,
   expressed as a decimal. A cap rate of 5% would be represented as 0.05.'
```

<floorRateSchedule> StrikeSchedule </floorRateSchedule> [0..*]

'The floor rate or floor rate schedule, if any, which applies to the floating rate. The floor rate (strike) is only required where the floating rate on a swap stream is floored at a certain strike level. A floor rate schedule is expressed as explicit floor rates and dates and the step dates may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments. The floor rate is assumed to be exclusive of any spread and is a per annum rate, expressed as a decimal. A floor rate of 5% would be represented as 0.05.'

<initialRate> xsd:decimal </initialRate> [0..1]

'The initial floating rate reset agreed between the principal parties involved in the trade. This is assumed to be the first required reset rate for the first regular calculation period. It should only be included when the rate is not equal to the rate published on the source implied by the floating rate index. An initial rate of 5% would be represented as 0.05.'

<finalRateRounding> Rounding </finalRateRounding> [0..1]

'The rounding convention to apply to the final rate used in determination of a calculation period amount.'

<averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]

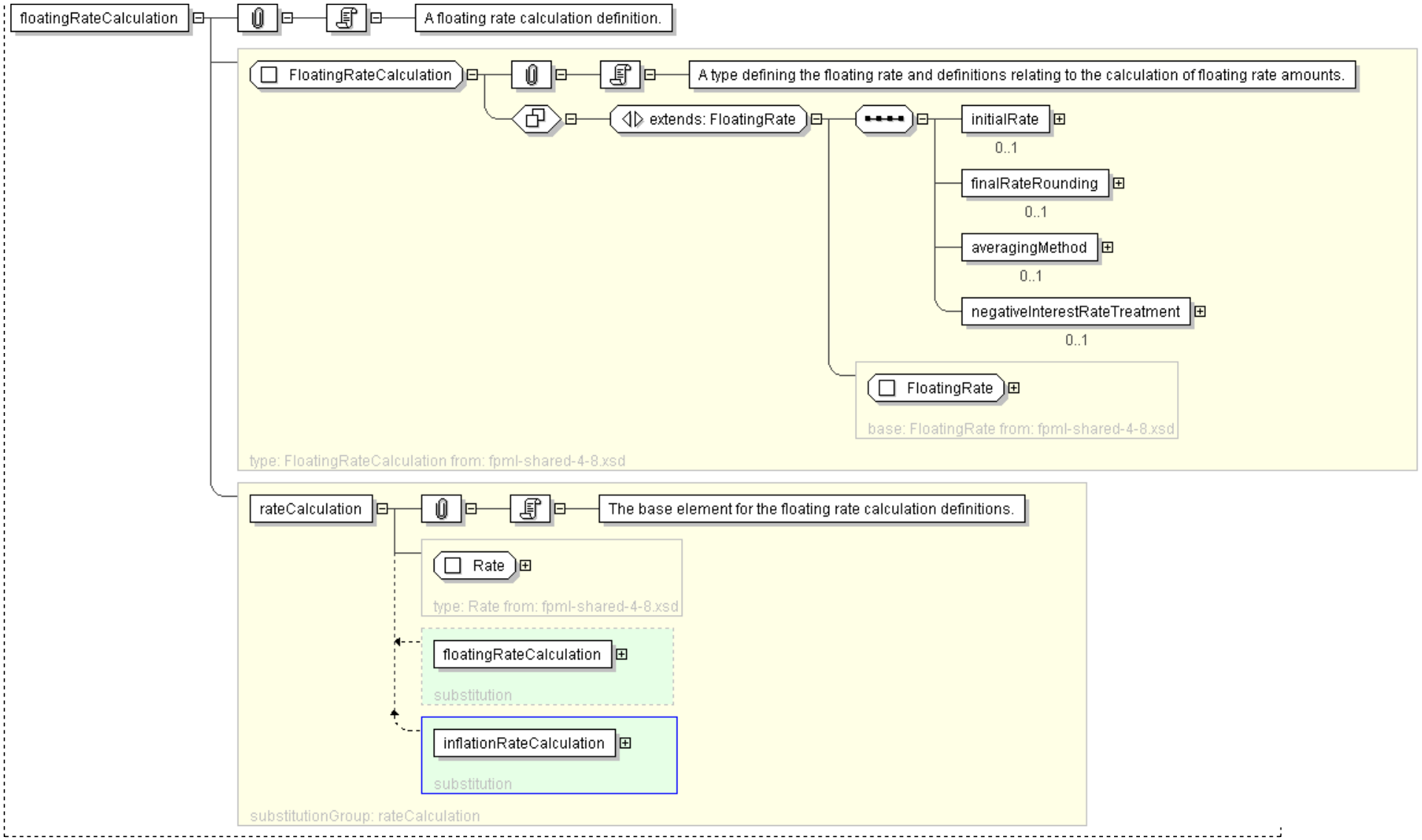
'If averaging is applicable, this component specifies whether a weighted or unweighted average method of calculation is to be used. The component must only be included when averaging applies.'

<negativeInterestRateTreatment> NegativeInterestRateTreatmentEnum </negativeInterestRateTreatment> [0..1]

'The specification of any provisions for calculating payment obligations when a floating rate is negative (either due to a quoted negative floating rate or by operation of a spread that is subtracted from the floating rate).'

</floatingRateCalculation>

Diagram



Schema Component Representation

```
<xsd:element name="floatingRateCalculation" type=" FloatingRateCalculation " substitutionGroup="rateCalculation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

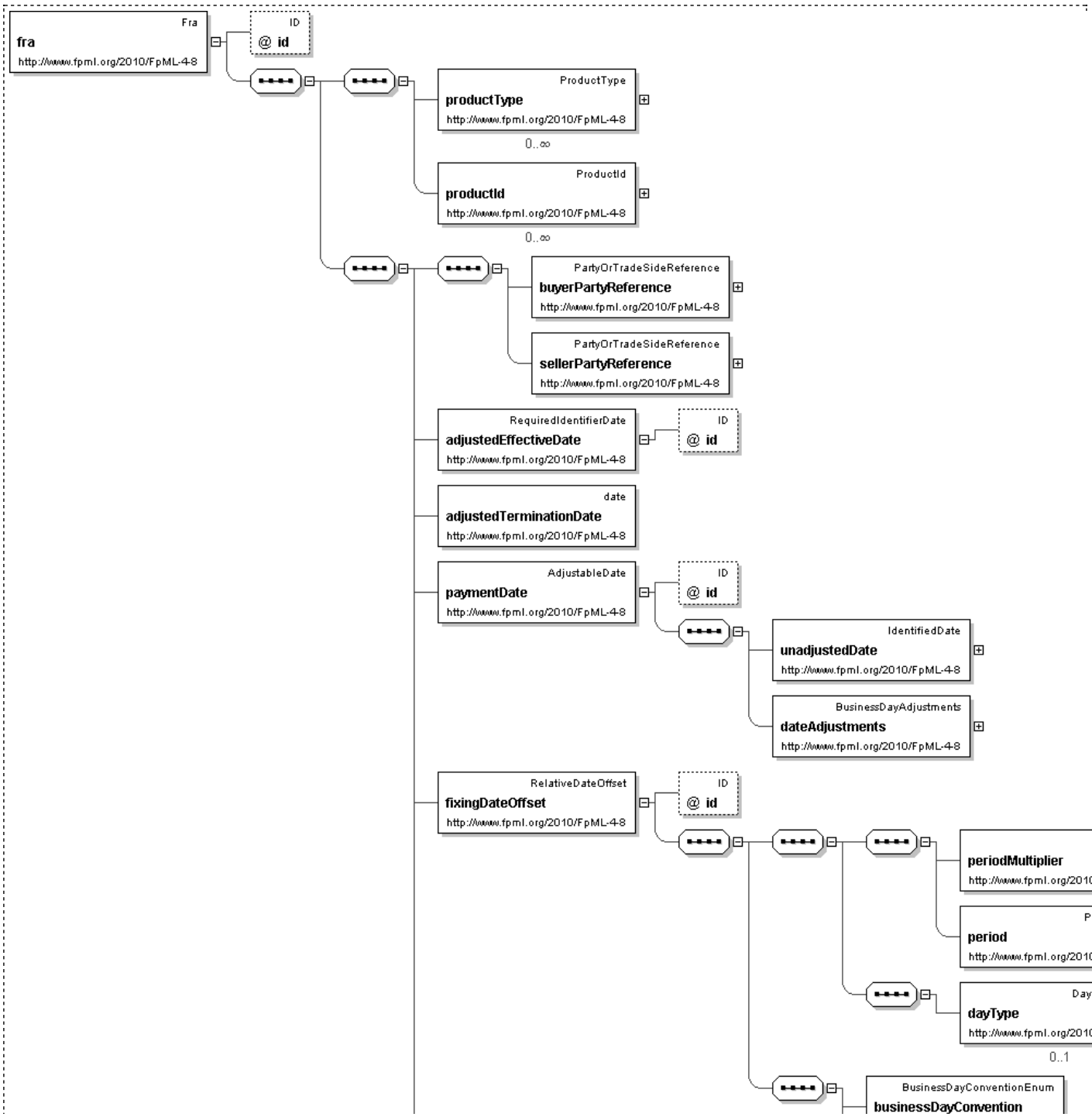
Element: fra

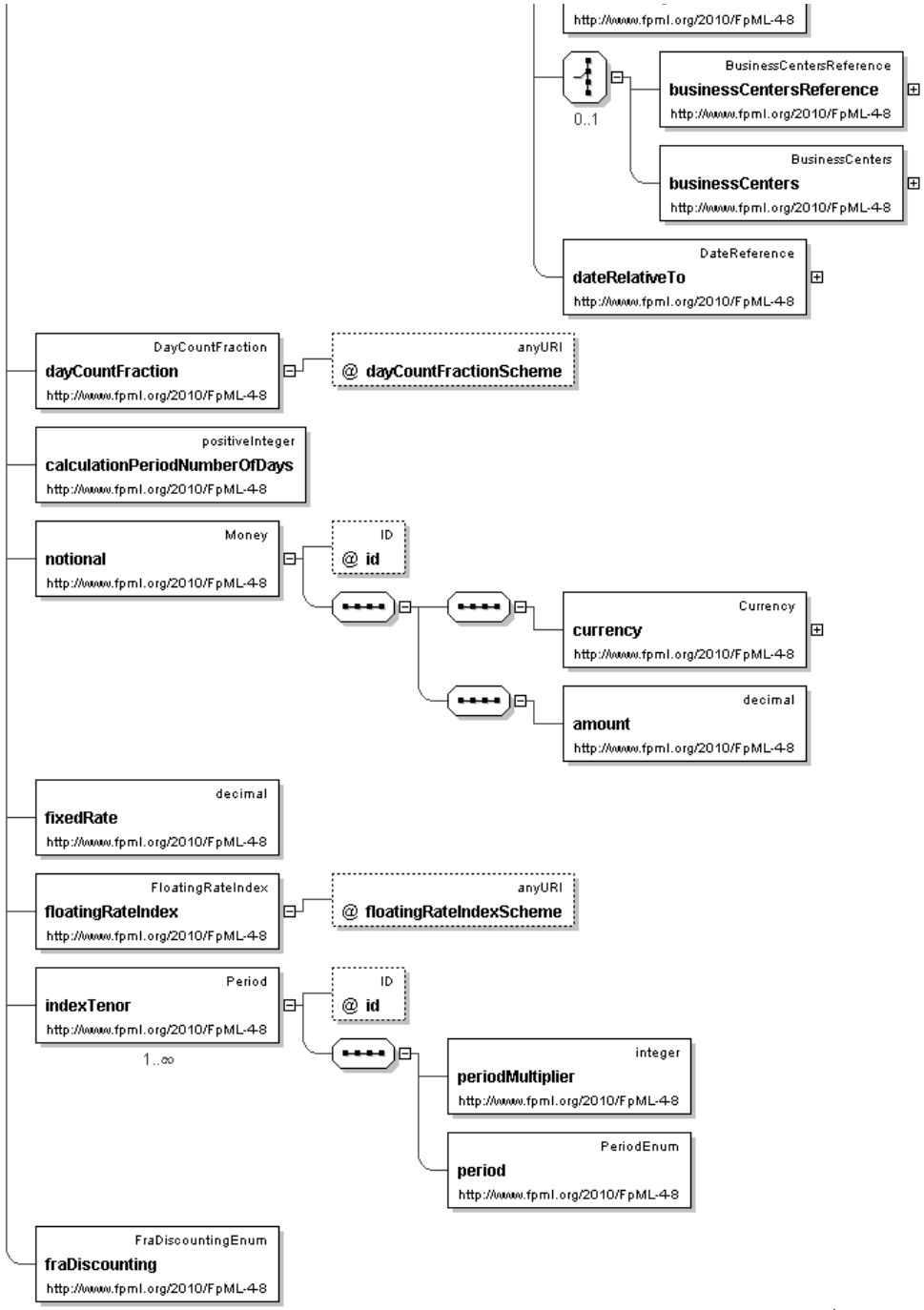
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	fra
Type	Fra
Nillable	no
Abstract	no
Documentation	A forward rate agreement product definition.

Logical Diagram





XML Instance Representation

```
<fra
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this
    element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by
    it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this
    instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs
    this is the floating rate payer.'

   <adjustedEffectiveDate> RequiredIdentifierDate </adjustedEffectiveDate> [1]
    'The start date of the calculation period. This date should already be adjusted for any applicable business day
    convention. This is also the date when the observed rate is applied, the reset date.'

   <adjustedTerminationDate> xsd:date </adjustedTerminationDate> [1]
    'The end date of the calculation period. This date should already be adjusted for any applicable business day
    convention.'

   <paymentDate> AdjustableDate </paymentDate> [1]
```

'The payment date. This date is subject to adjustment in accordance with any applicable business day convention.'

<fixingDateOffset> RelativeDateOffset </fixingDateOffset> [1]

'Specifies the fixing date relative to the reset date in terms of a business days offset and an associated set of financial business centers. Normally these offset calculation rules will be those specified in the ISDA definition for the relevant floating rate index (ISDA\'s Floating Rate Option). However, non-standard offset calculation rules may apply for a trade if mutually agreed by the principal parties to the transaction. The href attribute on the dateRelativeTo element should reference the id attribute on the adjustedEffectiveDate element.'

<dayCountFraction> DayCountFraction </dayCountFraction> [1]

'The day count fraction.'

<calculationPeriodNumberOfDays> xsd:positiveInteger </calculationPeriodNumberOfDays> [1]

'The number of days from the adjusted effective date to the adjusted termination date calculated in accordance with the applicable day count fraction.'

<notional> Money </notional> [1]

'The notional amount.'

<fixedRate> xsd:decimal </fixedRate> [1]

'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.'

<floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
<indexTenor> Period </indexTenor> [1..*]

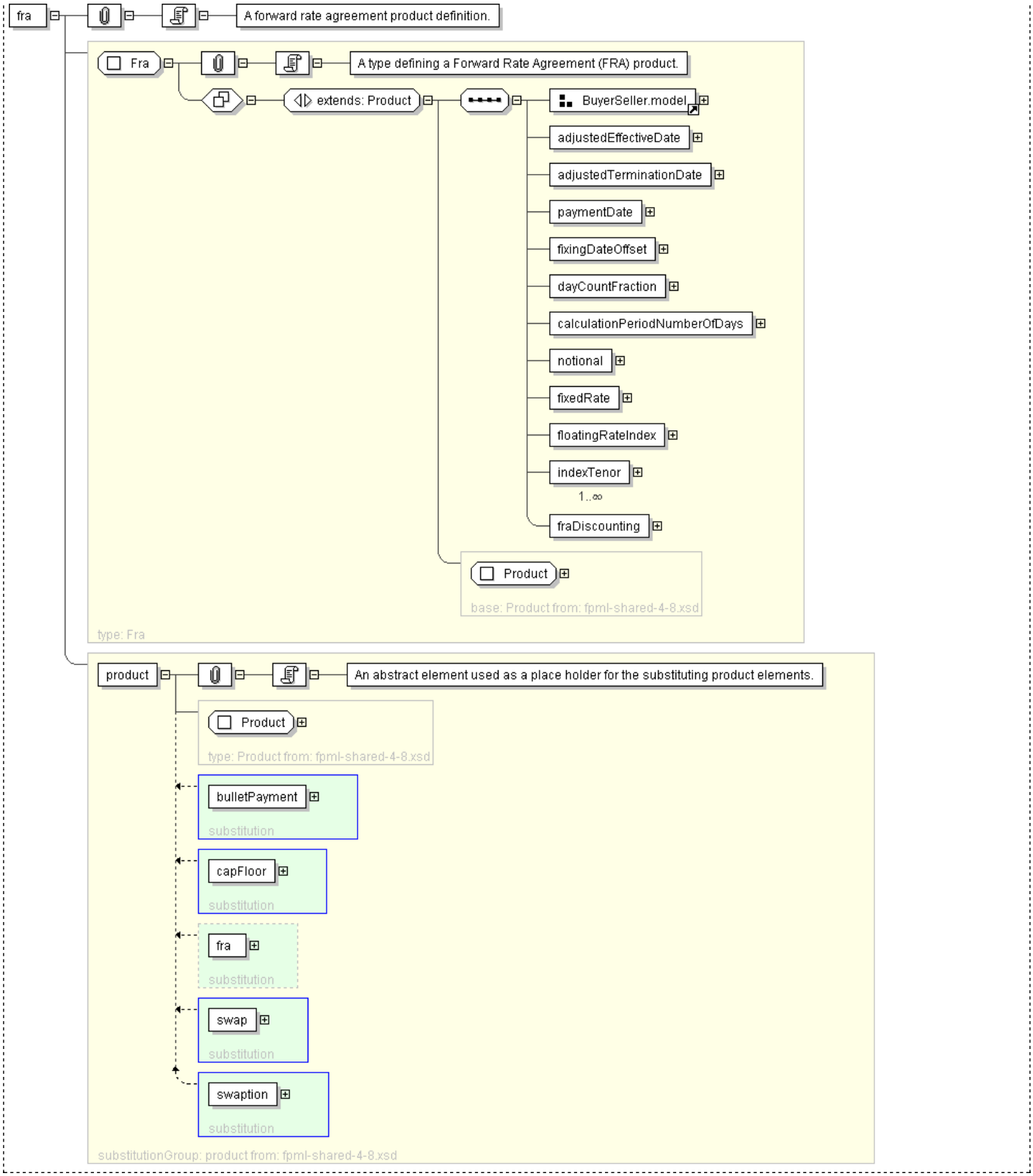
'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

<fraDiscounting> FraDiscountingEnum </fraDiscounting> [1]

'Specifies whether discounting applies and, if so, what type.'

</fra>

Diagram



Schema Component Representation

```
<xsd:element name="fra" type=" Fra " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

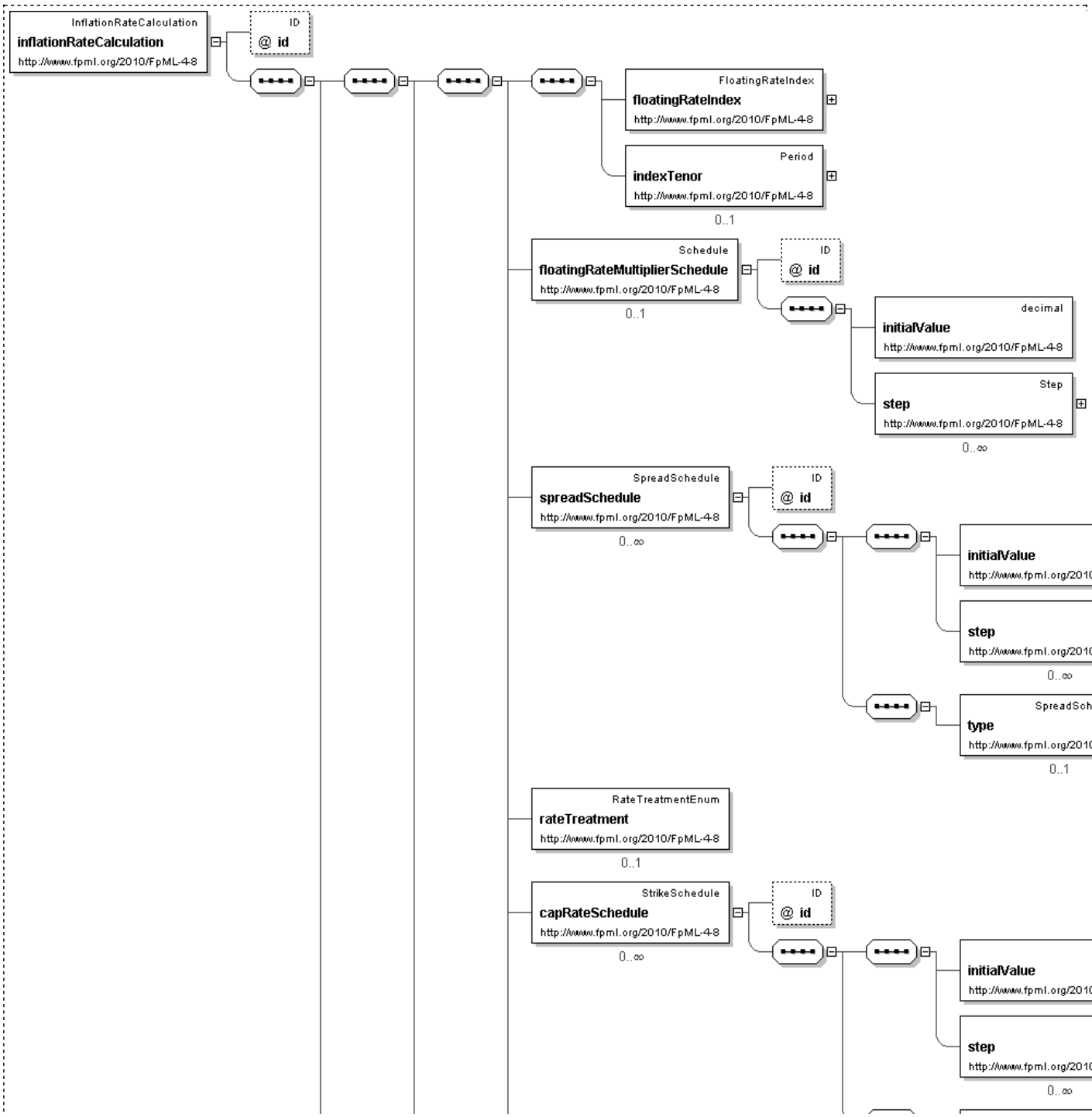
Element: inflationRateCalculation

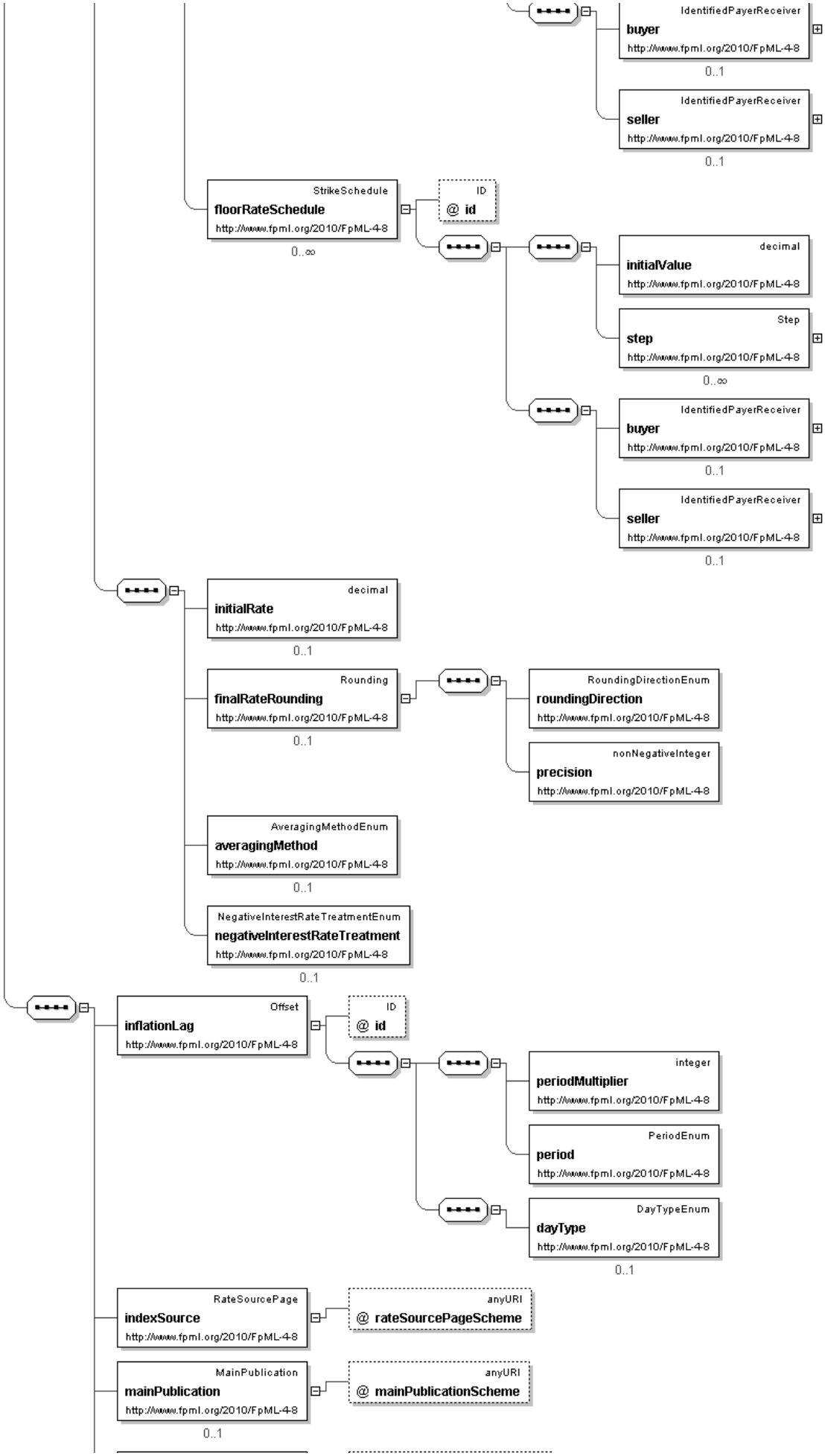
[Table of contents]

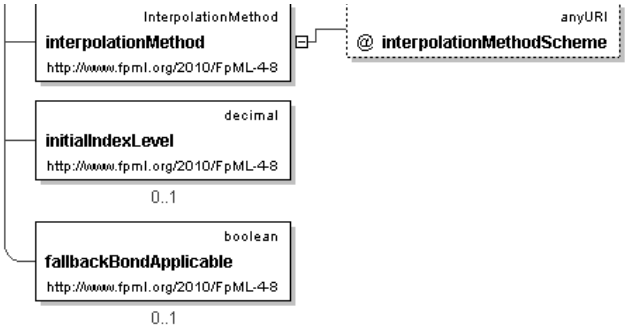
- *This element can be used wherever the following element is referenced:*
  - rateCalculation

Name	inflationRateCalculation
Type	InflationRateCalculation
Nillable	no
Abstract	no
Documentation	An inflation rate calculation definition.

Logical Diagram



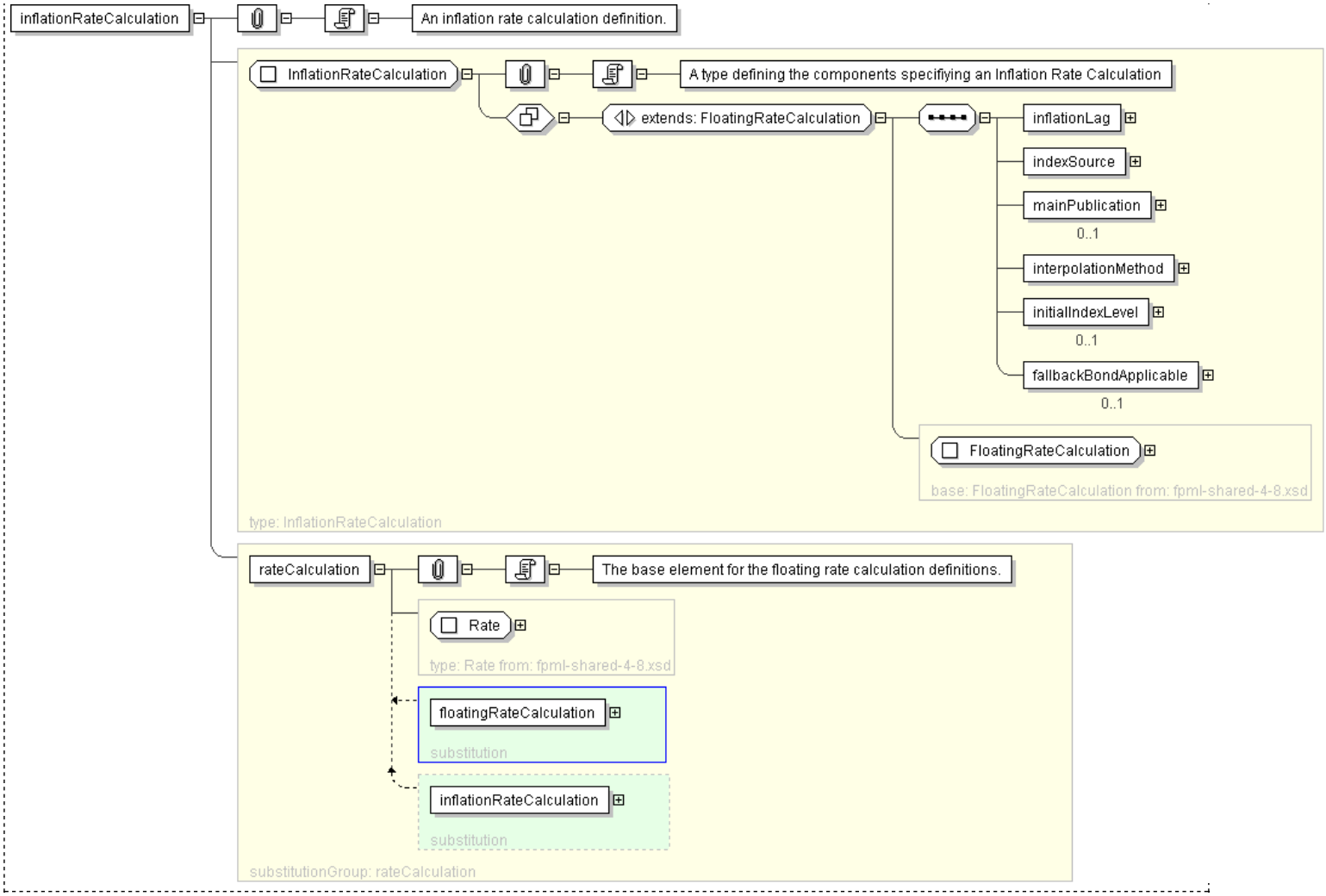

IdentifiedPayerReceiver
buyer
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedPayerReceiver
seller
http://www.fpml.org/2010/FpML-4-8
0..1
StrikeSchedule
floorRateSchedule
http://www.fpml.org/2010/FpML-4-8
0..∞
ID
@ id
decimal
initialValue
http://www.fpml.org/2010/FpML-4-8
Step
step
http://www.fpml.org/2010/FpML-4-8
0..∞
IdentifiedPayerReceiver
buyer
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedPayerReceiver
seller
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
initialRate
http://www.fpml.org/2010/FpML-4-8
0..1
Rounding
finalRateRounding
http://www.fpml.org/2010/FpML-4-8
0..1
RoundingDirectionEnum
roundingDirection
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
precision
http://www.fpml.org/2010/FpML-4-8
AveragingMethodEnum
averagingMethod
http://www.fpml.org/2010/FpML-4-8
0..1
NegativeInterestRateTreatmentEnum
negativeInterestRateTreatment
http://www.fpml.org/2010/FpML-4-8
0..1
Offset
inflationLag
http://www.fpml.org/2010/FpML-4-8
ID
@ id
integer
periodMultiplier
http://www.fpml.org/2010/FpML-4-8
PeriodEnum
period
http://www.fpml.org/2010/FpML-4-8
DayTypeEnum
dayType
http://www.fpml.org/2010/FpML-4-8
0..1
RateSourcePage
indexSource
http://www.fpml.org/2010/FpML-4-8
anyURI
@ rateSourcePageScheme
MainPublication
mainPublication
http://www.fpml.org/2010/FpML-4-8
0..1
anyURI
@ mainPublicationScheme



XML Instance Representation

```
<inflationRateCalculation
 id=" xsd:ID [0..1]">
  <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
  <indexTenor> Period </indexTenor> [0..1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

  <floatingRateMultiplierSchedule> Schedule </floatingRateMultiplierSchedule> [0..1]
   'A rate multiplier or multiplier schedule to apply to the floating rate. A multiplier schedule is expressed as
   explicit multipliers and dates. In the case of a schedule, the step dates may be subject to adjustment in accordance
   with any adjustments specified in the calculationPeriodDatesAdjustments. The multiplier can be a positive or negative
   decimal. This element should only be included if the multiplier is not equal to 1 (one) for the term of the stream.'

  <spreadSchedule> SpreadSchedule </spreadSchedule> [0..*]
   'The ISDA Spread or a Spread schedule expressed as explicit spreads and dates. In the case of a schedule, the step
   dates may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.
   The spread is a per annum rate, expressed as a decimal. For purposes of determining a calculation period amount, if
   positive the spread will be added to the floating rate and if negative the spread will be subtracted from the
   floating rate. A positive 10 basis point (0.1%) spread would be represented as 0.001.'

  <rateTreatment> RateTreatmentEnum </rateTreatment> [0..1]
   'The specification of any rate conversion which needs to be applied to the observed rate before being used in any
   calculations. The two common conversions are for securities quoted on a bank discount basis which will need to be
   converted to either a Money Market Yield or Bond Equivalent Yield. See the Annex to the 2000 ISDA Definitions,
   Section 7.3. Certain General Definitions Relating to Floating Rate Options, paragraphs (g) and (h) for definitions of
   these terms.'

  <capRateSchedule> StrikeSchedule </capRateSchedule> [0..*]
   'The cap rate or cap rate schedule, if any, which applies to the floating rate. The cap rate (strike) is only
   required where the floating rate on a swap stream is capped at a certain level. A cap rate schedule is expressed as
   explicit cap rates and dates and the step dates may be subject to adjustment in accordance with any adjustments
   specified in calculationPeriodDatesAdjustments. The cap rate is assumed to be exclusive of any spread and is a per
   annum rate, expressed as a decimal. A cap rate of 5% would be represented as 0.05.'

  <floorRateSchedule> StrikeSchedule </floorRateSchedule> [0..*]
   'The floor rate or floor rate schedule, if any, which applies to the floating rate. The floor rate (strike) is only
   required where the floating rate on a swap stream is floored at a certain strike level. A floor rate schedule is
   expressed as explicit floor rates and dates and the step dates may be subject to adjustment in accordance with any
   adjustments specified in calculationPeriodDatesAdjustments. The floor rate is assumed to be exclusive of any spread
   and is a per annum rate, expressed as a decimal. A floor rate of 5% would be represented as 0.05.'

  <initialRate> xsd:decimal </initialRate> [0..1]
   'The initial floating rate reset agreed between the principal parties involved in the trade. This is assumed to be
   the first required reset rate for the first regular calculation period. It should only be included when the rate is
   not equal to the rate published on the source implied by the floating rate index. An initial rate of 5% would be
   represented as 0.05.'

  <finalRateRounding> Rounding </finalRateRounding> [0..1]
   'The rounding convention to apply to the final rate used in determination of a calculation period amount.'

  <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
   'If averaging is applicable, this component specifies whether a weighted or unweighted average method of calculation
   is to be used. The component must only be included when averaging applies.'

  <negativeInterestRateTreatment> NegativeInterestRateTreatmentEnum </negativeInterestRateTreatment> [0..1]
   'The specification of any provisions for calculating payment obligations when a floating rate is negative (either due
   to a quoted negative floating rate or by operation of a spread that is subtracted from the floating rate).'

  <inflationLag> Offset </inflationLag> [1]
   'an offsetting period from the payment date which determines the reference period for which the inflation index is
   onserved.'

  <indexSource> RateSourcePage </indexSource> [1]
   'The reference source such as Reuters or Bloomberg.'

  <mainPublication> MainPublication </mainPublication> [0..1]
   'The current main publication source such as relevant web site or a government body.'

  <interpolationMethod> InterpolationMethod </interpolationMethod> [1]
   'The method used when calculating the Inflation Index Level from multiple points - the most common is Linear.'

  <initialIndexLevel> xsd:decimal </initialIndexLevel> [0..1]
   'initial known index level for the first calculation period.'

  <fallbackBondApplicable> xsd:boolean </fallbackBondApplicable> [0..1]
   'The applicability of a fallback bond as defined in the 2006 ISDA Inflation Derivatives Definitions, sections 1.3 and
   1.8. Omission of this element imples a value of true.'

</inflationRateCalculation>
```

Diagram



Schema Component Representation

```
<xsd:element name="inflationRateCalculation" type=" InflationRateCalculation " substitutionGroup="rateCalculation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

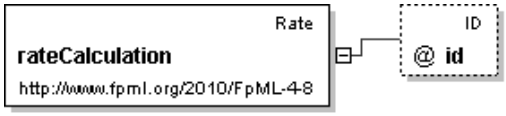
Element: rateCalculation

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - floatingRateCalculation
  - inflationRateCalculation

Name	rateCalculation
Used by (from the same schema document)	Complex Type Calculation
Type	Rate
Nillable	no
Abstract	yes
Documentation	The base element for the floating rate calculation definitions.

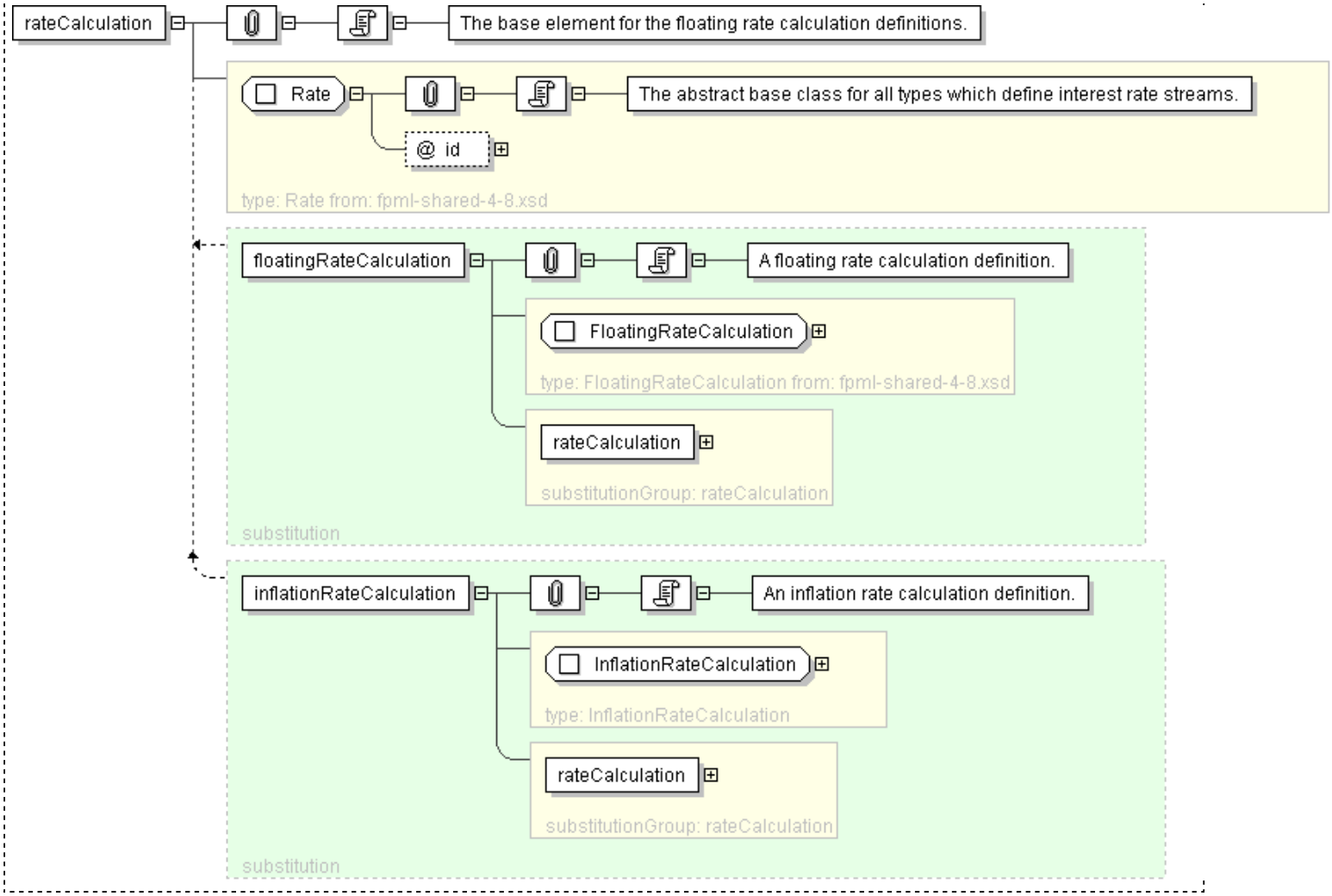
Logical Diagram



XML Instance Representation

```
<rateCalculation
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:element name="rateCalculation" type=" Rate " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

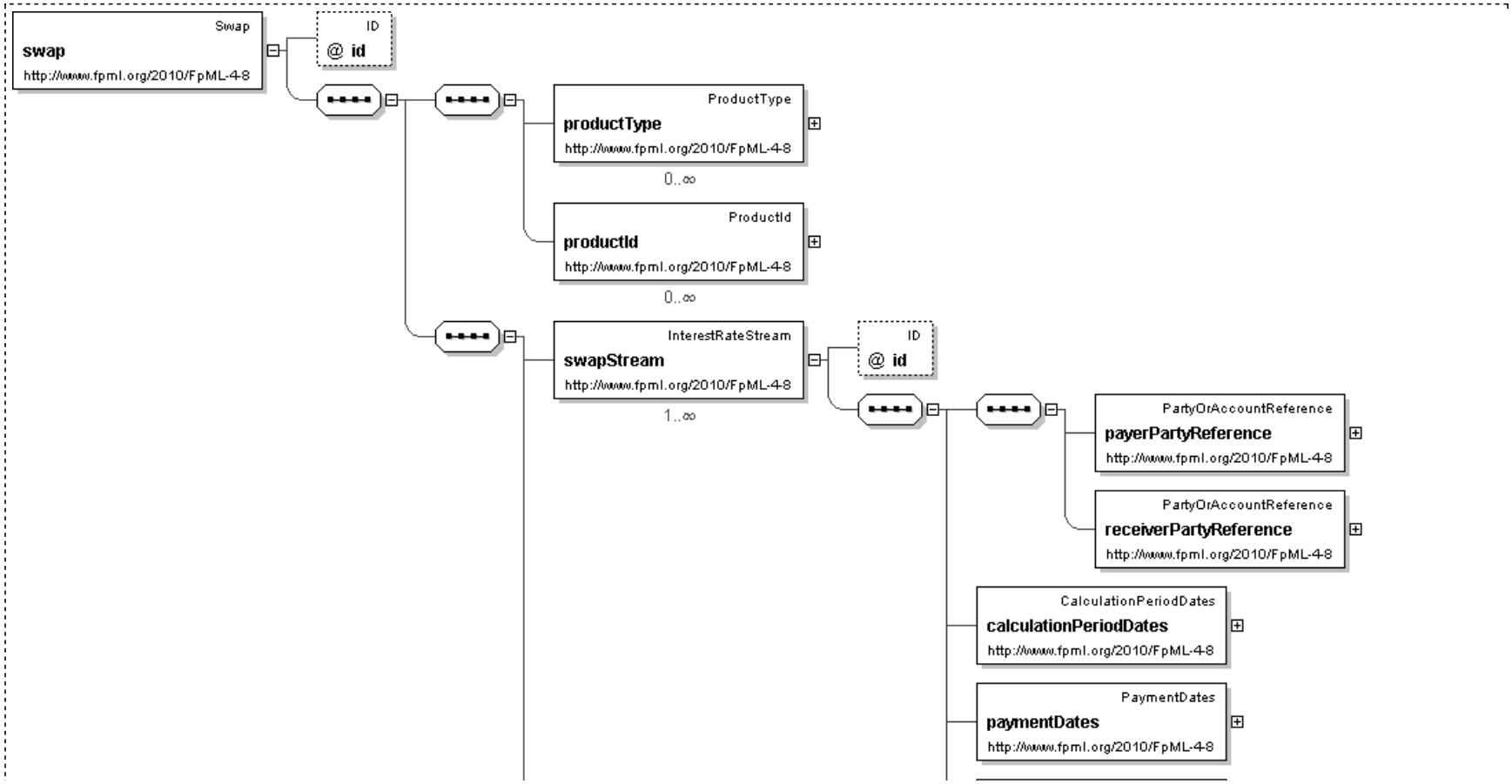
Element: swap

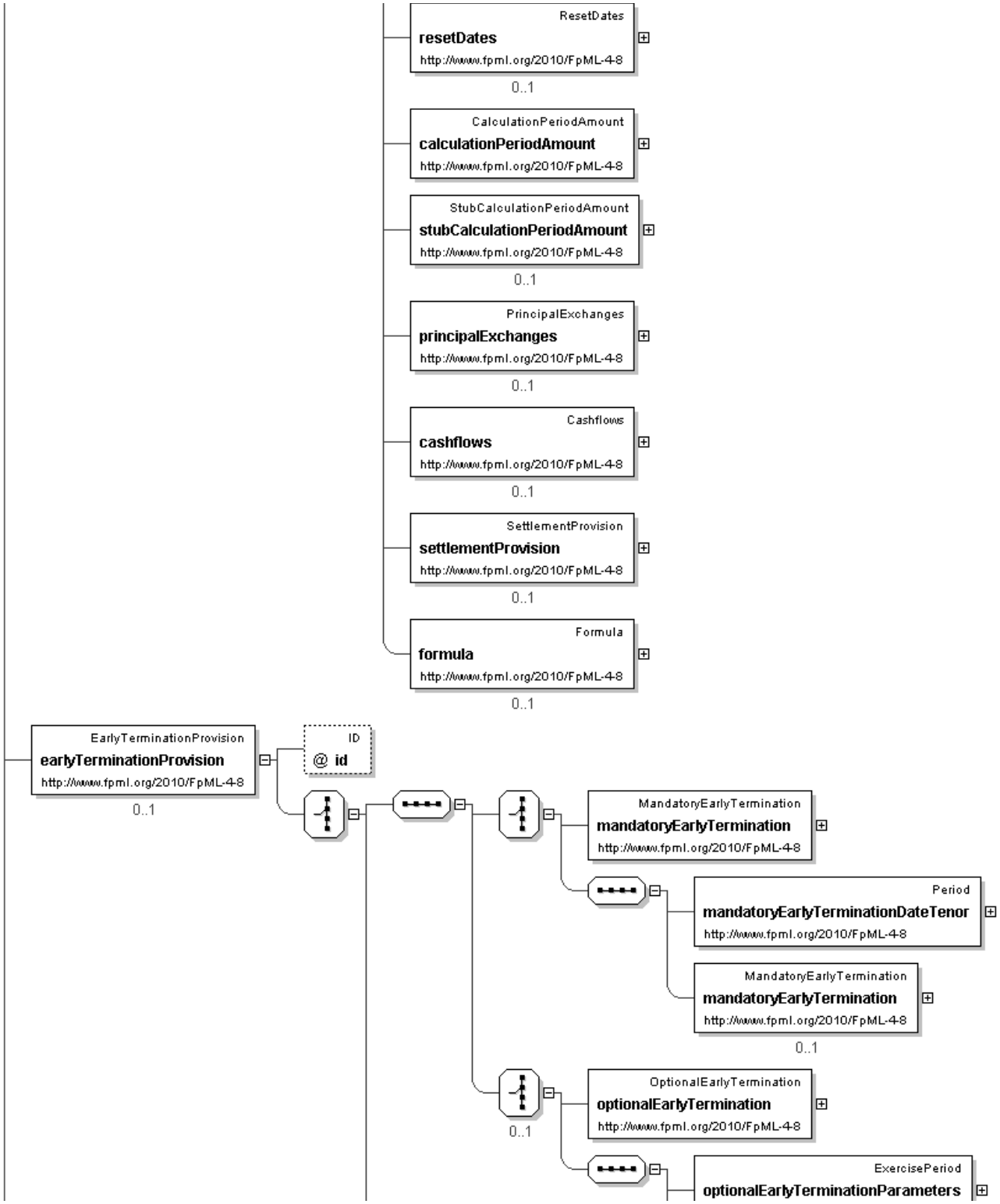
[Table of contents]

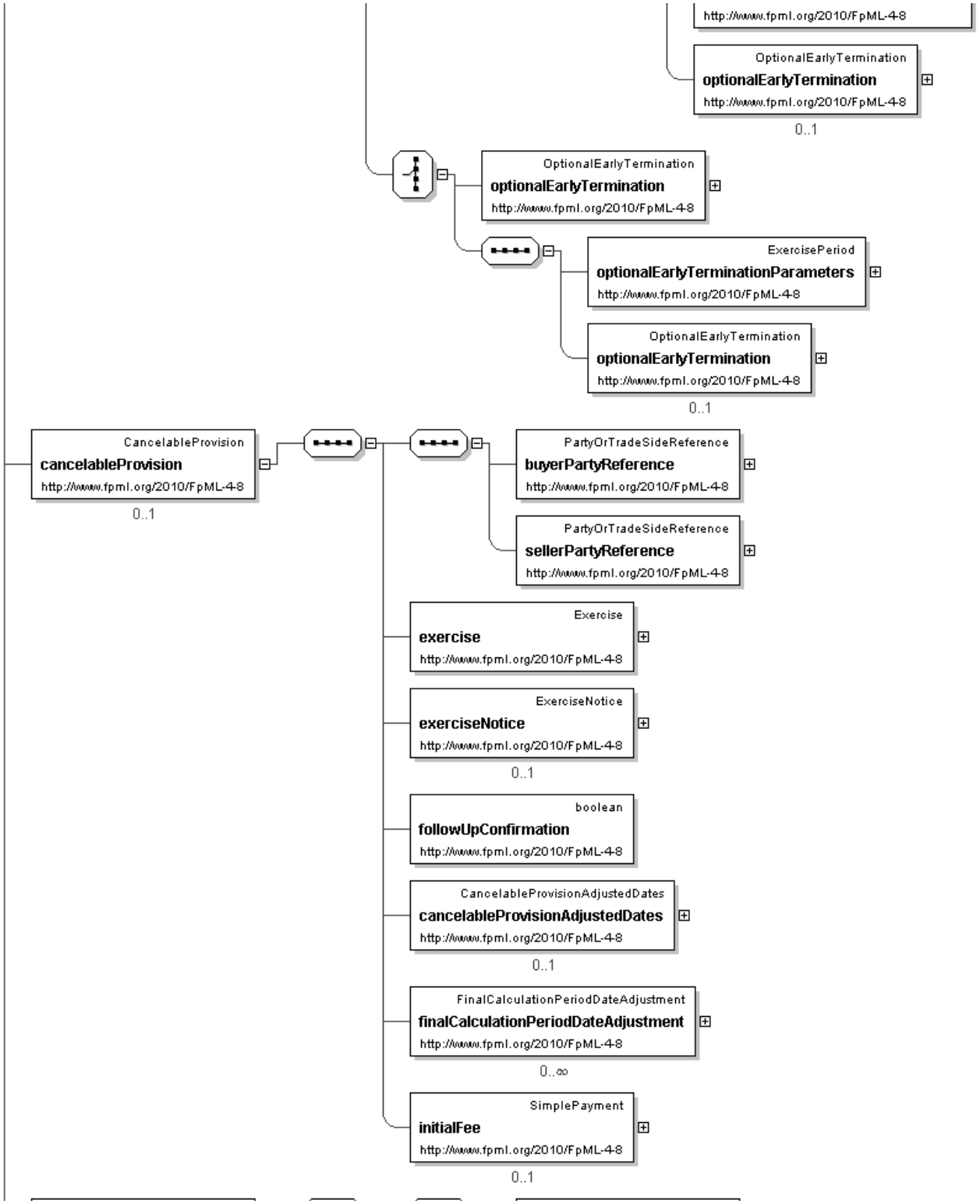
- *This element can be used wherever the following element is referenced:*
  - product

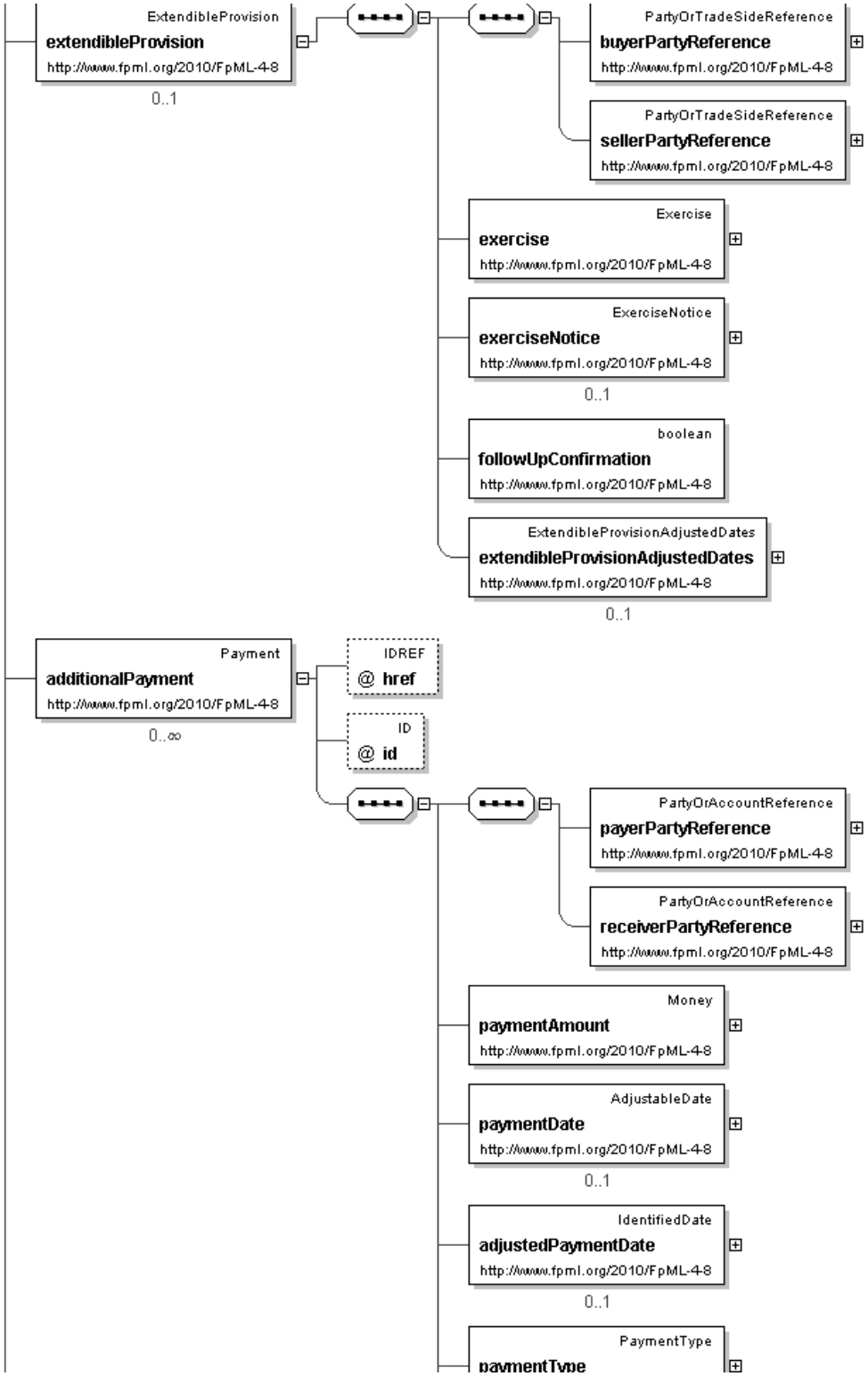
Name	swap
Used by (from the same schema document)	Complex Type Swaption
Type	Swap
Nillable	no
Abstract	no
Documentation	A swap product definition.

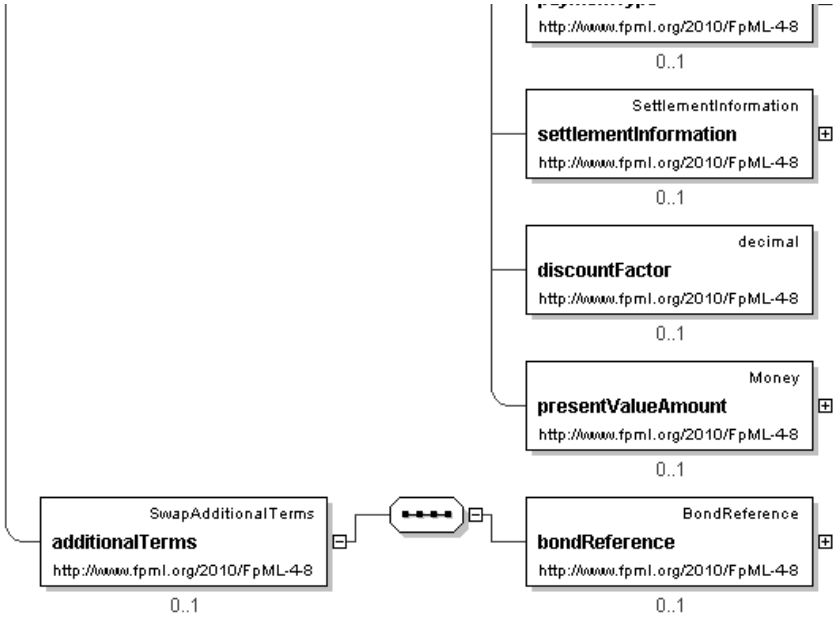
Logical Diagram




ResetDates
resetDates
http://www.fpml.org/2010/FpML-4-8
0..1
CalculationPeriodAmount
calculationPeriodAmount
http://www.fpml.org/2010/FpML-4-8
StubCalculationPeriodAmount
stubCalculationPeriodAmount
http://www.fpml.org/2010/FpML-4-8
0..1
PrincipalExchanges
principalExchanges
http://www.fpml.org/2010/FpML-4-8
0..1
Cashflows
cashflows
http://www.fpml.org/2010/FpML-4-8
0..1
SettlementProvision
settlementProvision
http://www.fpml.org/2010/FpML-4-8
0..1
Formula
formula
http://www.fpml.org/2010/FpML-4-8
0..1
EarlyTerminationProvision
earlyTerminationProvision
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
MandatoryEarlyTermination
mandatoryEarlyTermination
http://www.fpml.org/2010/FpML-4-8
Period
mandatoryEarlyTerminationDateTenor
http://www.fpml.org/2010/FpML-4-8
MandatoryEarlyTermination
mandatoryEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
ExercisePeriod
optionalEarlyTerminationParameters


http://www.fpml.org/2010/FpML-4-8
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
ExercisePeriod
optionalEarlyTerminationParameters
http://www.fpml.org/2010/FpML-4-8
OptionalEarlyTermination
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
CancelableProvision
cancelableProvision
http://www.fpml.org/2010/FpML-4-8
0..1
PartyOrTradeSideReference
buyerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrTradeSideReference
sellerPartyReference
http://www.fpml.org/2010/FpML-4-8
Exercise
exercise
http://www.fpml.org/2010/FpML-4-8
ExerciseNotice
exerciseNotice
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
followUpConfirmation
http://www.fpml.org/2010/FpML-4-8
CancelableProvisionAdjustedDates
cancelableProvisionAdjustedDates
http://www.fpml.org/2010/FpML-4-8
0..1
FinalCalculationPeriodDateAdjustment
finalCalculationPeriodDateAdjustment
http://www.fpml.org/2010/FpML-4-8
0..∞
SimplePayment
initialFee
http://www.fpml.org/2010/FpML-4-8
0..1

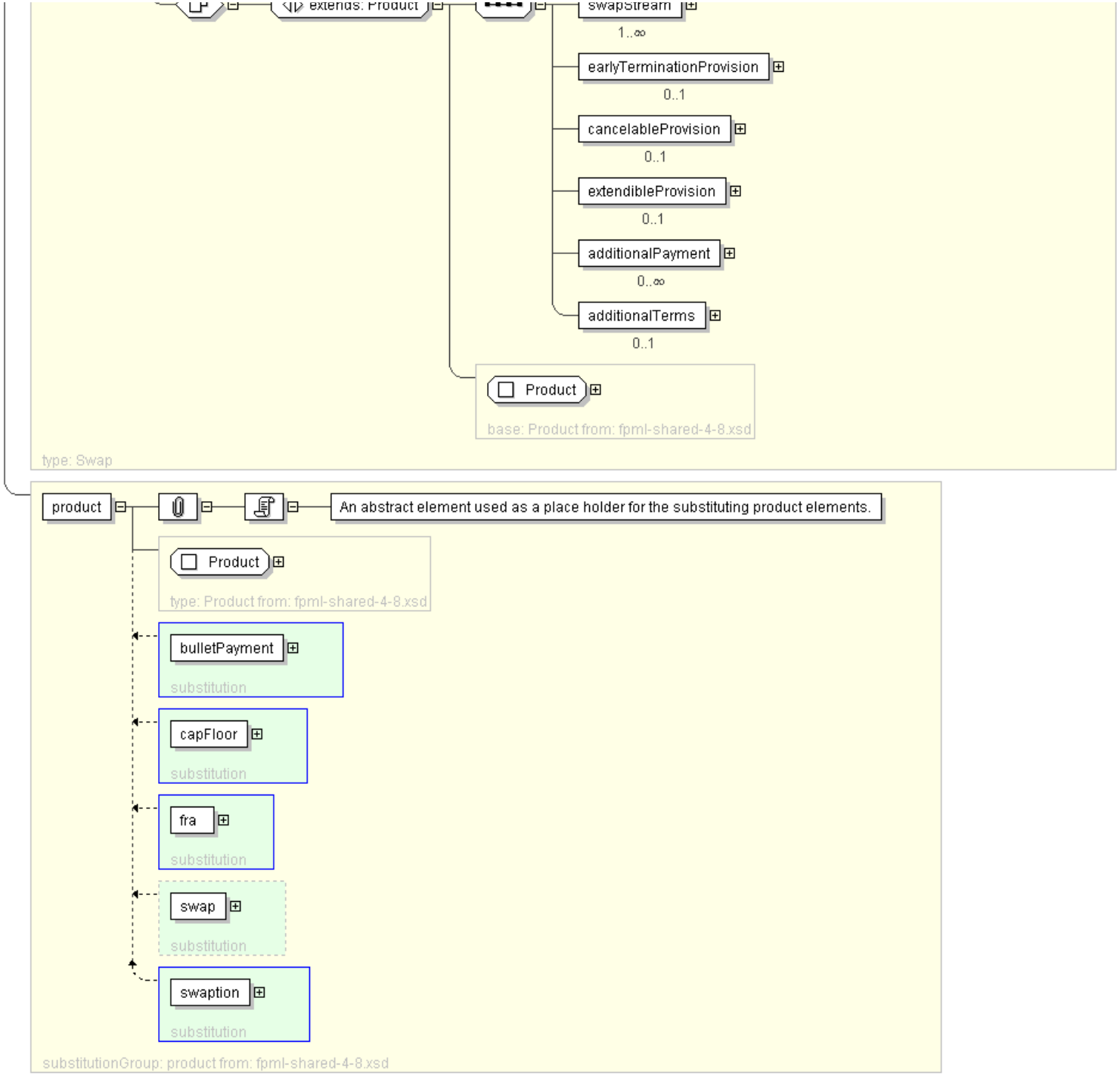

ExtendibleProvision
extendibleProvision
http://www.fpml.org/2010/FpML-4-8
0..1
PartyOrTradeSideReference
buyerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrTradeSideReference
sellerPartyReference
http://www.fpml.org/2010/FpML-4-8
Exercise
exercise
http://www.fpml.org/2010/FpML-4-8
ExerciseNotice
exerciseNotice
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
followUpConfirmation
http://www.fpml.org/2010/FpML-4-8
ExtendibleProvisionAdjustedDates
extendibleProvisionAdjustedDates
http://www.fpml.org/2010/FpML-4-8
0..1
Payment
additionalPayment
http://www.fpml.org/2010/FpML-4-8
0..∞
IDREF
@ href
ID
@ id
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
Money
paymentAmount
http://www.fpml.org/2010/FpML-4-8
AdjustableDate
paymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
adjustedPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
PaymentType
paymentType



XML Instance Representation

```
<swap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
   that the domain values for this element are not strictly an enumerated list.'

   <swapStream> InterestRateStream </swapStream> [1..*]
   'The swap streams.'

   <earlyTerminationProvision> EarlyTerminationProvision </earlyTerminationProvision> [0..1]
   'Parameters specifying provisions relating to the optional and mandatory early terminarion of a swap transaction.'

   <cancelableProvision> CancelableProvision </cancelableProvision> [0..1]
   'A provision that allows the specification of an embedded option within a swap giving the buyer of the option the right to
   terminate the swap, in whole or in part, on the early termination date.'

   <extendibleProvision> ExtendibleProvision </extendibleProvision> [0..1]
   'A provision that allows the specification of an embedded option with a swap giving the buyer of the option the right to extend
   the swap, in whole or in part, to the extended termination date.'

   <additionalPayment> Payment </additionalPayment> [0..*]
   'Additional payments between the principal parties.'

   <additionalTerms> SwapAdditionalTerms </additionalTerms> [0..1]
   'Contains any additional terms to the swap contract.'

</swap>
```

Diagram



Schema Component Representation

```
<xsd:element name="swap" type=" Swap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

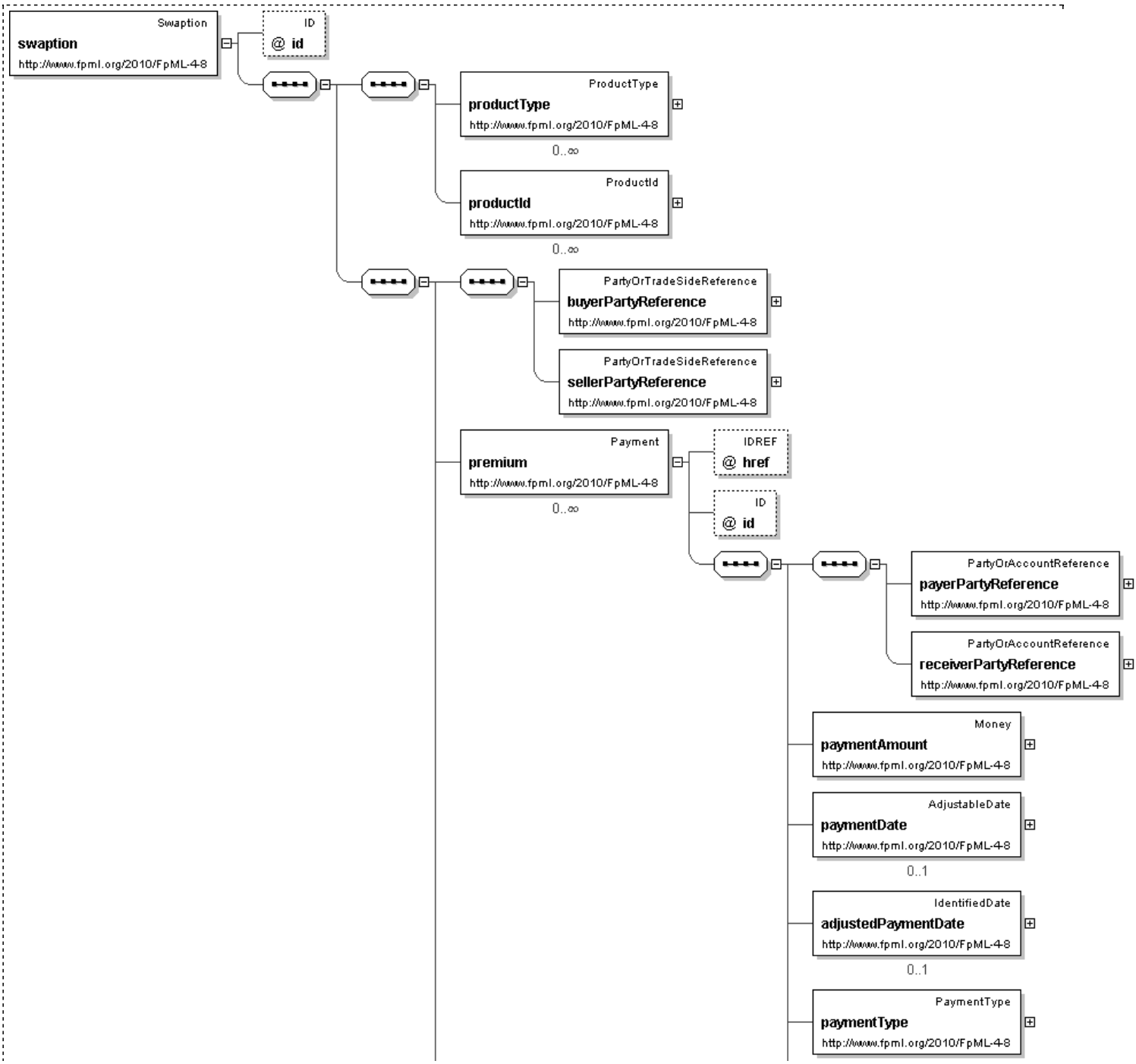
Element: swaption

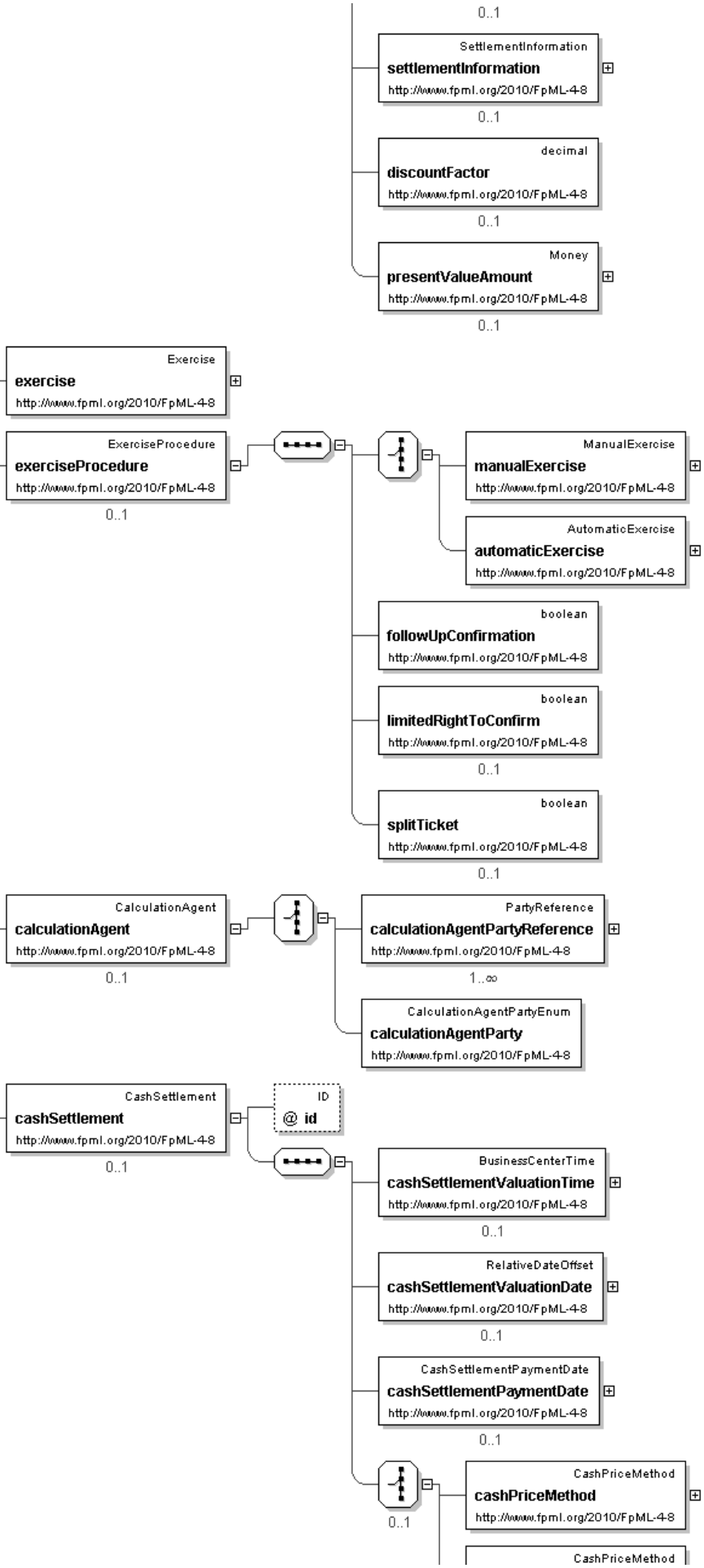
[Table of contents]

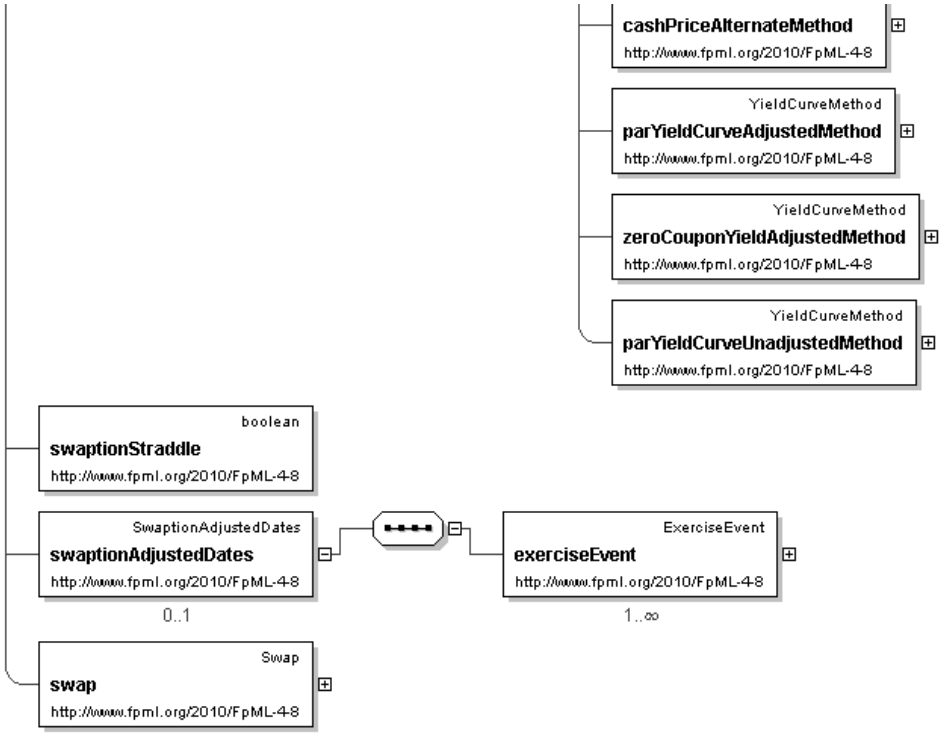
- *This element can be used wherever the following element is referenced:*
  - product

Name	swaption
Type	Swaption
Nillable	no
Abstract	no
Documentation	A swaption product definition.

Logical Diagram



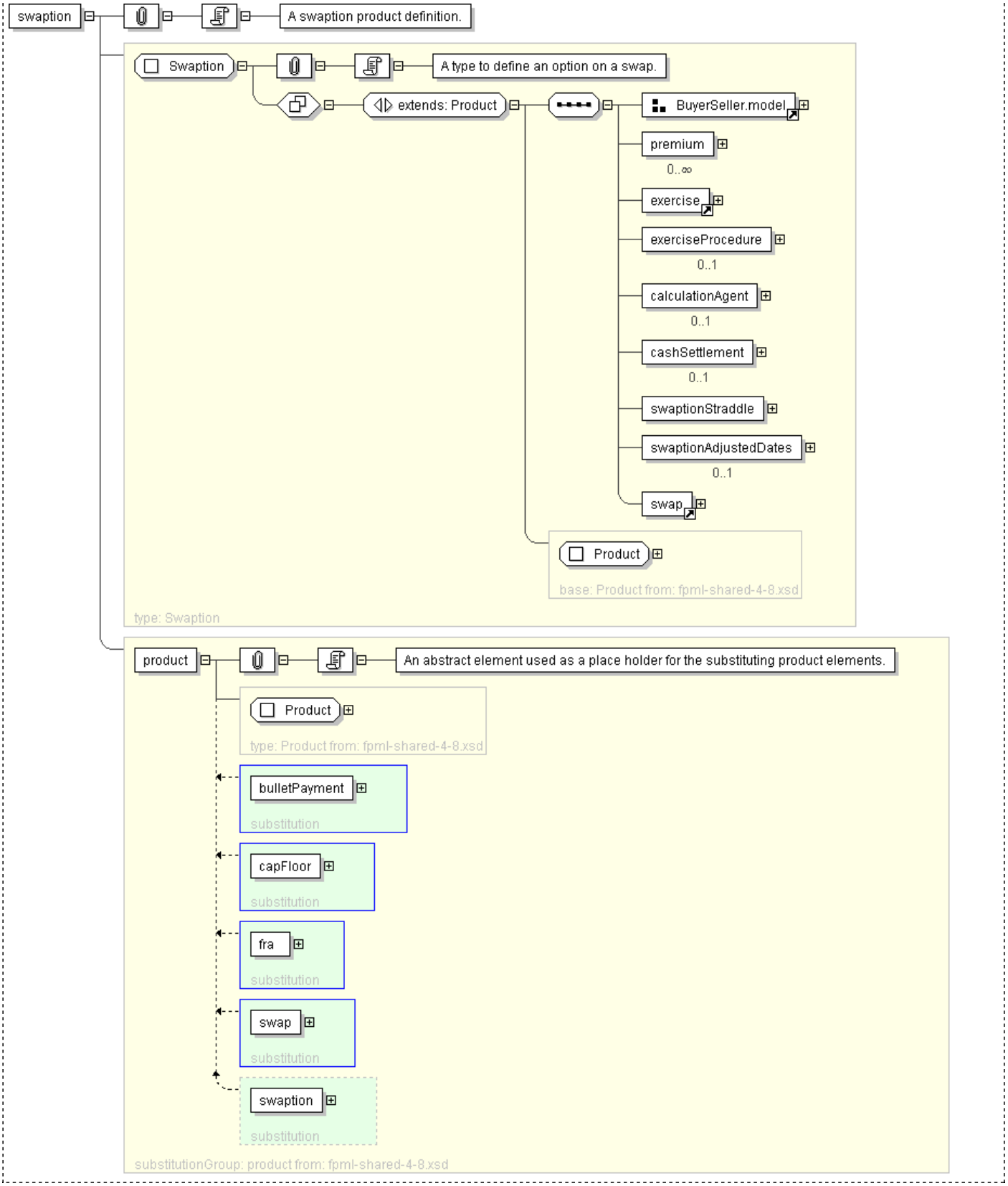

0..1
SettlementInformation
settlementInformation
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
discountFactor
http://www.fpml.org/2010/FpML-4-8
0..1
Money
presentValueAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Exercise
exercise
http://www.fpml.org/2010/FpML-4-8
ExerciseProcedure
exerciseProcedure
http://www.fpml.org/2010/FpML-4-8
0..1
ManualExercise
manualExercise
http://www.fpml.org/2010/FpML-4-8
AutomaticExercise
automaticExercise
http://www.fpml.org/2010/FpML-4-8
boolean
followUpConfirmation
http://www.fpml.org/2010/FpML-4-8
boolean
limitedRightToConfirm
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
splitTicket
http://www.fpml.org/2010/FpML-4-8
0..1
CalculationAgent
calculationAgent
http://www.fpml.org/2010/FpML-4-8
0..1
PartyReference
calculationAgentPartyReference
http://www.fpml.org/2010/FpML-4-8
1..∞
CalculationAgentPartyEnum
calculationAgentParty
http://www.fpml.org/2010/FpML-4-8
CashSettlement
cashSettlement
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
BusinessCenterTime
cashSettlementValuationTime
http://www.fpml.org/2010/FpML-4-8
0..1
RelativeDateOffset
cashSettlementValuationDate
http://www.fpml.org/2010/FpML-4-8
0..1
CashSettlementPaymentDate
cashSettlementPaymentDate
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
CashPriceMethod
cashPriceMethod
http://www.fpml.org/2010/FpML-4-8
CashPriceMethod



XML Instance Representation

```
<swaption
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with
   this element. Note that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
   defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by
   this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the
   case of FRAs this is the floating rate payer.'

   <premium> Payment </premium> [0..*]
   'The option premium amount payable by buyer to seller on the specified payment date.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [0..1]
   'A set of parameters defining procedures associated with the exercise.'

   <calculationAgent> CalculationAgent </calculationAgent> [0..1]
   'The ISDA Calculation Agent responsible for performing duties associated with an optional early
   termination.'

   <cashSettlement> CashSettlement </cashSettlement> [0..1]
   'If specified, this means that cash settlement is applicable to the transaction and defines the parameters
   associated with the cash settlement prodcedure. If not specified, then physical settlement is applicable.'

   <swaptionStraddle> xsd:boolean </swaptionStraddle> [1]
   'Whether the option is a swaption or a swaption straddle.'

   <swaptionAdjustedDates> SwaptionAdjustedDates </swaptionAdjustedDates> [0..1]
   'The adjusted dates associated with swaption exercise. These dates have been adjusted for any applicable
   business day convention.'

   <swap> ... </swap> [1]
</swaption>
```

Diagram



Schema Component Representation

```
<xsd:element name="swaption" type=" Swaption " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: DiscountRate.model

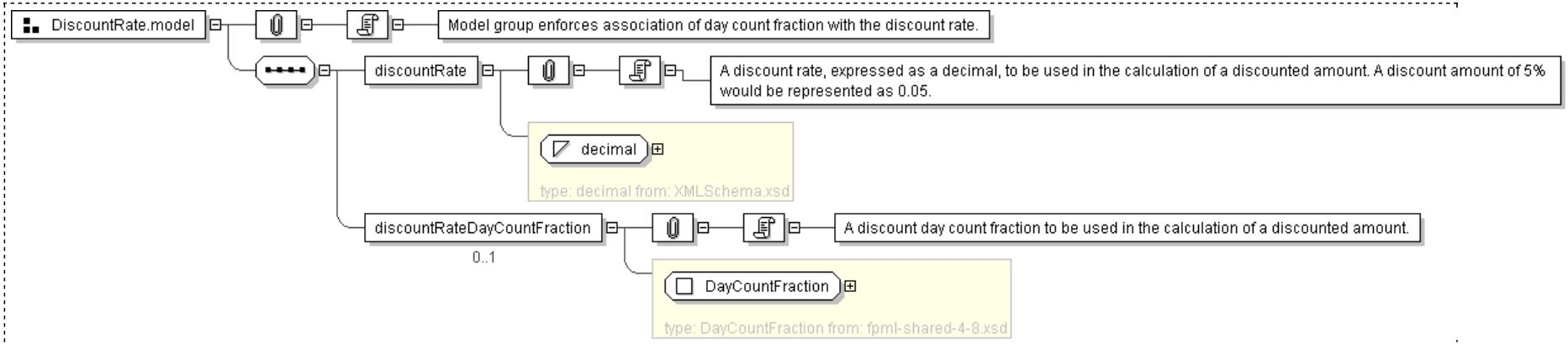
[Table of contents]

Name	DiscountRate.model
Used by (from the same schema document)	Complex Type Discounting
Documentation	Model group enforces association of day count fraction with the discount rate.

XML Instance Representation

```
<discountRate> xsd:decimal </discountRate> [1]
'A discount rate, expressed as a decimal, to be used in the calculation of a discounted amount. A discount amount of 5% would be
represented as 0.05.'

<discountRateDayCountFraction> DayCountFraction </discountRateDayCountFraction> [0..1]
'A discount day count fraction to be used in the calculation of a discounted amount.'
```

Diagram



Schema Component Representation

```
<xsd:group name="DiscountRate.model">
  <xsd:sequence>
    <xsd:element name="discountRate" type=" xsd:decimal "/>
    <xsd:element name="discountRateDayCountFraction" type=" DayCountFraction " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: MandatoryEarlyTermination.model

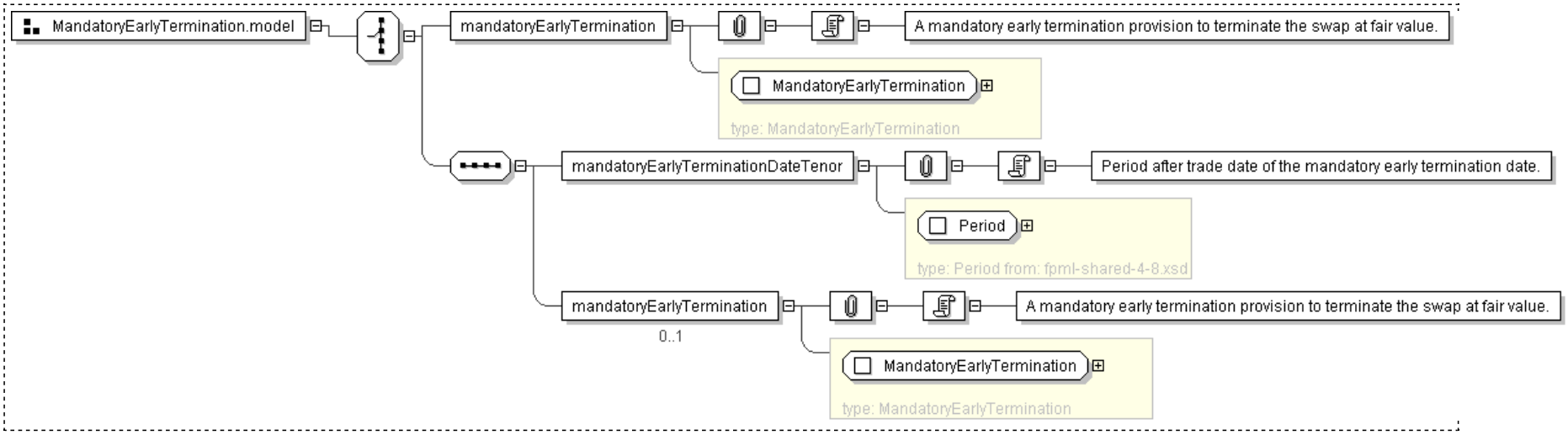
[Table of contents]

Name	MandatoryEarlyTermination.model
Used by (from the same schema document)	Complex Type EarlyTerminationProvision

XML Instance Representation

```
Start Choice [1]
   <mandatoryEarlyTermination> MandatoryEarlyTermination </mandatoryEarlyTermination> [1]
   'A mandatory early termination provision to terminate the swap at fair value.'

   <mandatoryEarlyTerminationDateTenor> Period </mandatoryEarlyTerminationDateTenor> [1]
   'Period after trade date of the mandatory early termination date.'

   <mandatoryEarlyTermination> MandatoryEarlyTermination </mandatoryEarlyTermination> [0..1]
   'A mandatory early termination provision to terminate the swap at fair value.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="MandatoryEarlyTermination.model">
  <xsd:choice>
    <xsd:element name="mandatoryEarlyTermination" type=" MandatoryEarlyTermination "/>
    <xsd:sequence>
      <xsd:element name="mandatoryEarlyTerminationDateTenor" type=" Period "/>
      <xsd:element name="mandatoryEarlyTermination" type=" MandatoryEarlyTermination " minOccurs="0"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: OptionalEarlyTermination.model

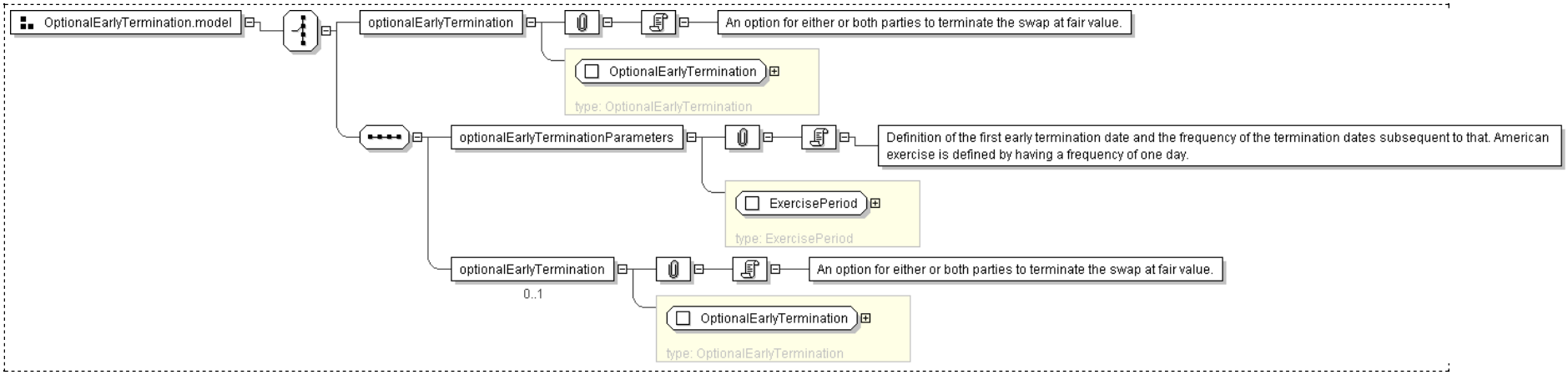
[Table of contents]

Name	OptionalEarlyTermination.model
Used by (from the same schema document)	Complex Type EarlyTerminationProvision , Complex Type EarlyTerminationProvision

XML Instance Representation

```
Start Choice [1]
   <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [1]
   'An option for either or both parties to terminate the swap at fair value.'

   <optionalEarlyTerminationParameters> ExercisePeriod </optionalEarlyTerminationParameters> [1]
   'Definition of the first early termination date and the frequency of the termination dates subsequent to that. American exercise is defined by having a frequency
   of one day.'

   <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [0..1]
   'An option for either or both parties to terminate the swap at fair value.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="OptionalEarlyTermination.model">
   <xsd:choice>
      <xsd:element name="optionalEarlyTermination" type=" OptionalEarlyTermination "/>
      <xsd:sequence>
         <xsd:element name="optionalEarlyTerminationParameters" type=" ExercisePeriod "/>
         <xsd:element name="optionalEarlyTermination" type=" OptionalEarlyTermination " minOccurs="0"/>
      </xsd:sequence>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BondReference

[Table of contents]

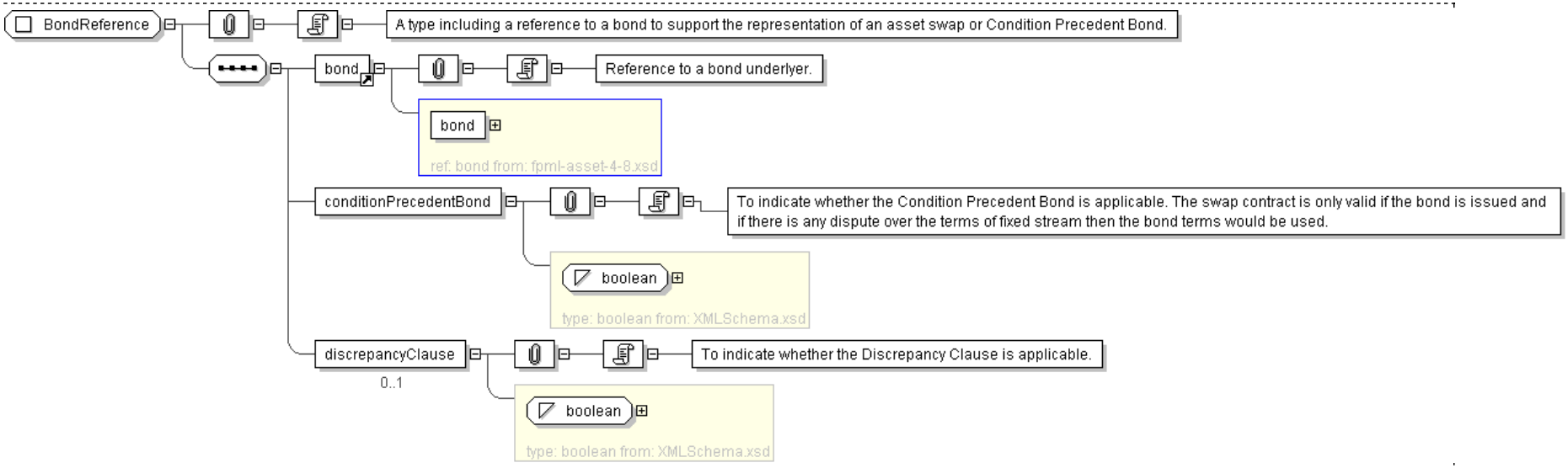
Super-types:	None
Sub-types:	None

Name	BondReference
Used by (from the same schema document)	Complex Type SwapAdditionalTerms
Abstract	no
Documentation	A type including a reference to a bond to support the representation of an asset swap or Condition Precedent Bond.

XML Instance Representation

```
<...>
  <bond> ... </bond> [1]
  'Reference to a bond underlyer.'

  <conditionPrecedentBond> xsd:boolean </conditionPrecedentBond> [1]
  'To indicate whether the Condition Precedent Bond is applicable. The swap contract is only valid if the bond is issued and if there is any
  dispute over the terms of fixed stream then the bond terms would be used.'

  <discrepancyClause> xsd:boolean </discrepancyClause> [0..1]
  'To indicate whether the Discrepancy Clause is applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BondReference">
  <xsd:sequence>
    <xsd:element ref=" bond "/>
    <xsd:element name="conditionPrecedentBond" type=" xsd:boolean "/>
    <xsd:element name="discrepancyClause" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BulletPayment

[Table of contents]

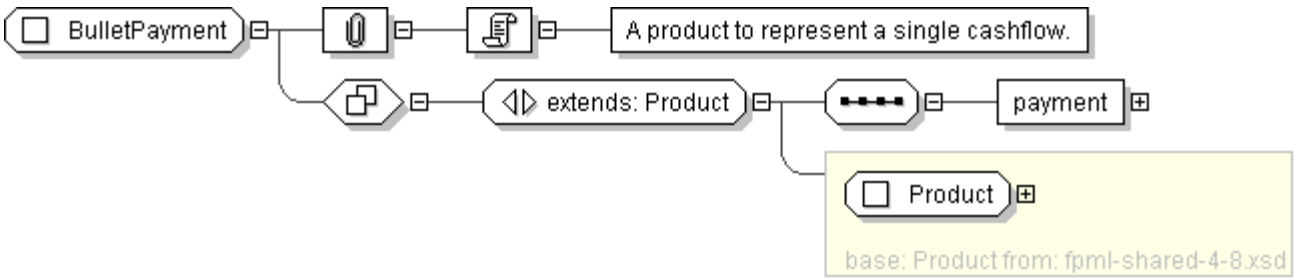
Super-types:	Product < BulletPayment (by extension)
Sub-types:	None

Name	BulletPayment
Used by (from the same schema document)	Element bulletPayment
Abstract	no
Documentation	A product to represent a single cashflow.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <payment> Payment </payment> [1]
    'A known payment between two parties.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BulletPayment">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="payment" type=" Payment "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Calculation

[Table of contents]

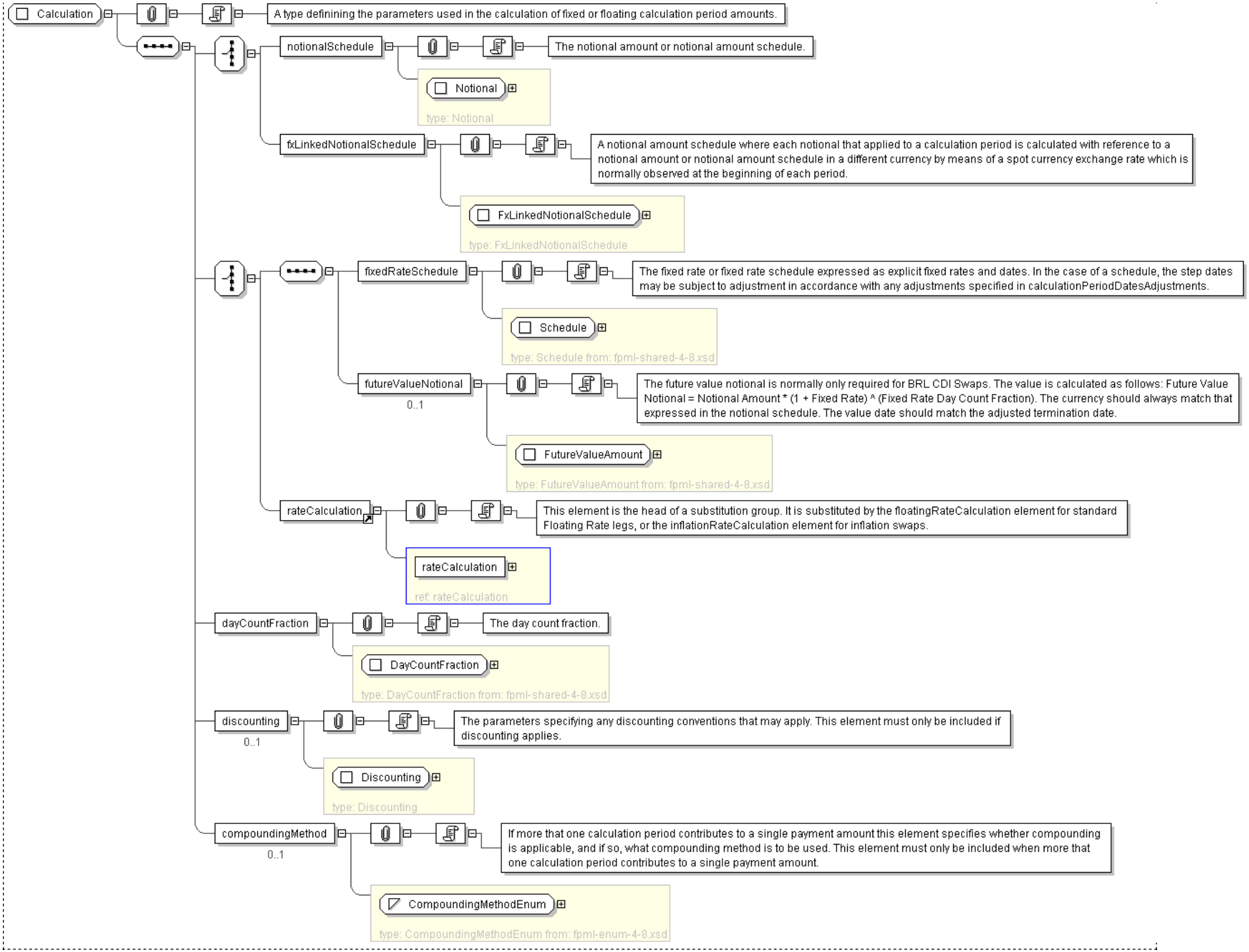
Super-types:	None
Sub-types:	None

Name	Calculation
Used by (from the same schema document)	Complex Type CalculationPeriodAmount
Abstract	no
Documentation	A type definining the parameters used in the calculation of fixed or floating calculation period amounts.

XML Instance Representation

```
<...>
   Start Choice [1]
      <notionalSchedule> Notional </notionalSchedule> [1]
      'The notional amount or notional amount schedule.'

      <fxLinkedNotionalSchedule> FxLinkedNotionalSchedule </fxLinkedNotionalSchedule> [1]
      'A notional amount schedule where each notional that applied to a calculation period is calculated with reference to a notional amount or notional
      amount schedule in a different currency by means of a spot currency exchange rate which is normally observed at the beginning of each period.'

   End Choice
   Start Choice [1]
      <fixedRateSchedule> Schedule </fixedRateSchedule> [1]
      'The fixed rate or fixed rate schedule expressed as explicit fixed rates and dates. In the case of a schedule, the step dates may be subject to
      adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.'

      <futureValueNotional> FutureValueAmount </futureValueNotional> [0..1]
      'The future value notional is normally only required for BRL CDI Swaps. The value is calculated as follows: Future Value Notional = Notional Amount *
      (1 + Fixed Rate) ^ (Fixed Rate Day Count Fraction). The currency should always match that expressed in the notional schedule. The value date should
      match the adjusted termination date.'

      <rateCalculation> ... </rateCalculation> [1]
      'This element is the head of a substitution group. It is substituted by the floatingRateCalculation element for standard Floating Rate legs, or the
      inflationRateCalculation element for inflation swaps.'

   End Choice
   <dayCountFraction> DayCountFraction </dayCountFraction> [1]
   'The day count fraction.'

   <discounting> Discounting </discounting> [0..1]
   'The parameters specifying any discounting conventions that may apply. This element must only be included if discounting applies.'

   <compoundingMethod> CompoundingMethodEnum </compoundingMethod> [0..1]
   'If more that one calculation period contributes to a single payment amount this element specifies whether compounding is applicable, and if so, what
   compounding method is to be used. This element must only be included when more that one calculation period contributes to a single payment amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Calculation">
  <xsd:sequence>
    <xsd:choice>
```

```
        <xsd:element name="notionalSchedule" type=" Notional "/>
        <xsd:element name="fxLinkedNotionalSchedule" type=" FxLinkedNotionalSchedule "/>
      </xsd:choice>
      <xsd:choice>
        <xsd:sequence>
          <xsd:element name="fixedRateSchedule" type=" Schedule "/>
          <xsd:element name="futureValueNotional" type=" FutureValueAmount " minOccurs="0"/>
        </xsd:sequence>
        <xsd:element ref=" rateCalculation "/>
      </xsd:choice>
      <xsd:element name="dayCountFraction" type=" DayCountFraction "/>
      <xsd:element name="discounting" type=" Discounting " minOccurs="0"/>
      <xsd:element name="compoundingMethod" type=" CompoundingMethodEnum " minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriod

[Table of contents]

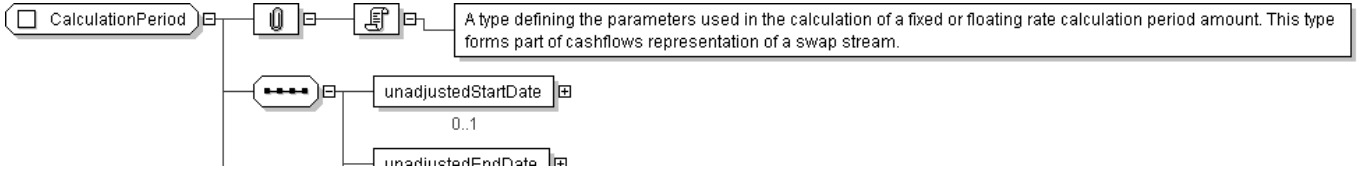
Super-types:	None
Sub-types:	None

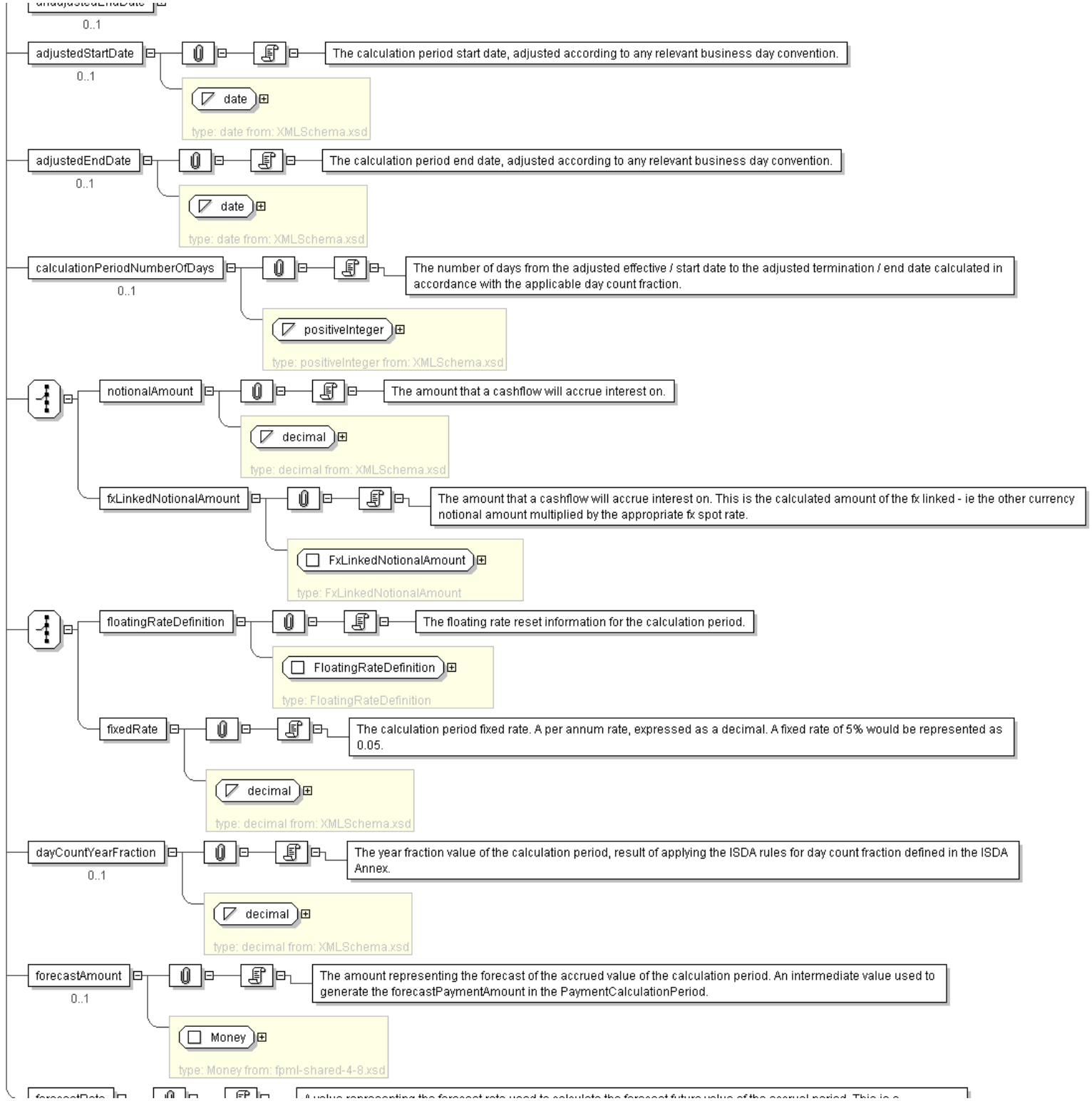
Name	CalculationPeriod
Used by (from the same schema document)	Complex Type PaymentCalculationPeriod
Abstract	no
Documentation	A type defining the parameters used in the calculation of a fixed or floating rate calculation period amount. This type forms part of cashflows representation of a swap stream.

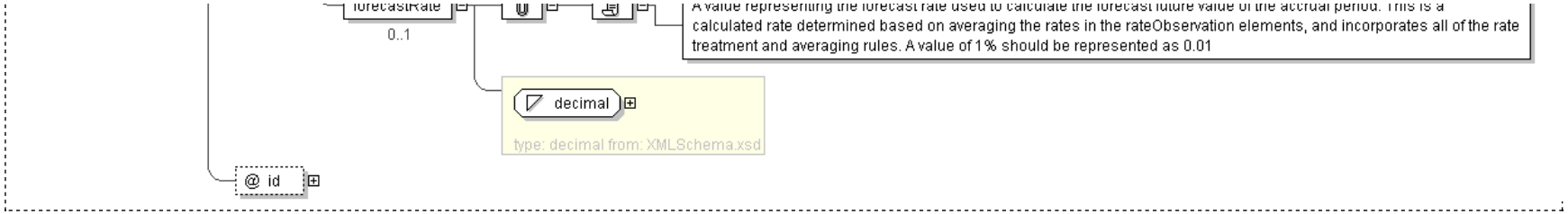
XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedStartDate> xsd:date </unadjustedStartDate> [0..1]
   <unadjustedEndDate> xsd:date </unadjustedEndDate> [0..1]
   <adjustedStartDate> xsd:date </adjustedStartDate> [0..1]
    'The calculation period start date, adjusted according to any relevant business day convention.'

   <adjustedEndDate> xsd:date </adjustedEndDate> [0..1]
    'The calculation period end date, adjusted according to any relevant business day convention.'

   <calculationPeriodNumberOfDays> xsd:positiveInteger </calculationPeriodNumberOfDays> [0..1]
    'The number of days from the adjusted effective / start date to the adjusted termination / end date calculated in accordance with the applicable day
    count fraction.'

   Start Choice [1]
      <notionalAmount> xsd:decimal </notionalAmount> [1]
       'The amount that a cashflow will accrue interest on.'

      <fxLinkedNotionalAmount> FxLinkedNotionalAmount </fxLinkedNotionalAmount> [1]
       'The amount that a cashflow will accrue interest on. This is the calculated amount of the fx linked - ie the other currency notional amount
       multiplied by the appropriate fx spot rate.'

   End Choice
   Start Choice [1]
      <floatingRateDefinition> FloatingRateDefinition </floatingRateDefinition> [1]
       'The floating rate reset information for the calculation period.'

      <fixedRate> xsd:decimal </fixedRate> [1]
       'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.'

   End Choice
   <dayCountYearFraction> xsd:decimal </dayCountYearFraction> [0..1]
    'The year fraction value of the calculation period, result of applying the ISDA rules for day count fraction defined in the ISDA Annex.'

   <forecastAmount> Money </forecastAmount> [0..1]
    'The amount representing the forecast of the accrued value of the calculation period. An intermediate value used to generate the
    forecastPaymentAmount in the PaymentCalculationPeriod.'

   <forecastRate> xsd:decimal </forecastRate> [0..1]
    'A value representing the forecast rate used to calculate the forecast future value of the accrual period. This is a calculated rate determined
    based on averaging the rates in the rateObservation elements, and incorporates all of the rate treatment and averaging rules. A value of 1% should
    be represented as 0.01'

</...>
```

Diagram




0..1
adjustedStartDate
0..1
The calculation period start date, adjusted according to any relevant business day convention.
date
type: date from: XMLSchema.xsd
adjustedEndDate
0..1
The calculation period end date, adjusted according to any relevant business day convention.
date
type: date from: XMLSchema.xsd
calculationPeriodNumberOfDays
0..1
The number of days from the adjusted effective / start date to the adjusted termination / end date calculated in accordance with the applicable day count fraction.
positiveInteger
type: positiveInteger from: XMLSchema.xsd
notionalAmount
The amount that a cashflow will accrue interest on.
decimal
type: decimal from: XMLSchema.xsd
fxLinkedNotionalAmount
The amount that a cashflow will accrue interest on. This is the calculated amount of the fx linked - ie the other currency notional amount multiplied by the appropriate fx spot rate.
FxLinkedNotionalAmount
type: FxLinkedNotionalAmount
floatingRateDefinition
The floating rate reset information for the calculation period.
FloatingRateDefinition
type: FloatingRateDefinition
fixedRate
The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.
decimal
type: decimal from: XMLSchema.xsd
dayCountYearFraction
0..1
The year fraction value of the calculation period, result of applying the ISDA rules for day count fraction defined in the ISDA Annex.
decimal
type: decimal from: XMLSchema.xsd
forecastAmount
0..1
The amount representing the forecast of the accrued value of the calculation period. An intermediate value used to generate the forecastPaymentAmount in the PaymentCalculationPeriod.
Money
type: Money from: fpml-shared-4-8.xsd



Schema Component Representation

```
<xsd:complexType name="CalculationPeriod">
   <xsd:sequence>
      <xsd:element name="unadjustedStartDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="unadjustedEndDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="adjustedStartDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="adjustedEndDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="calculationPeriodNumberOfDays" type=" xsd:positiveInteger " minOccurs="0"/>
      <xsd:choice>
         <xsd:element name="notionalAmount" type=" xsd:decimal "/>
         <xsd:element name="fxLinkedNotionalAmount" type=" FxLinkedNotionalAmount "/>
      </xsd:choice>
      <xsd:choice>
         <xsd:element name="floatingRateDefinition" type=" FloatingRateDefinition "/>
         <xsd:element name="fixedRate" type=" xsd:decimal "/>
      </xsd:choice>
      <xsd:element name="dayCountYearFraction" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="forecastAmount" type=" Money " minOccurs="0"/>
      <xsd:element name="forecastRate" type=" xsd:decimal " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodAmount

[Table of contents]

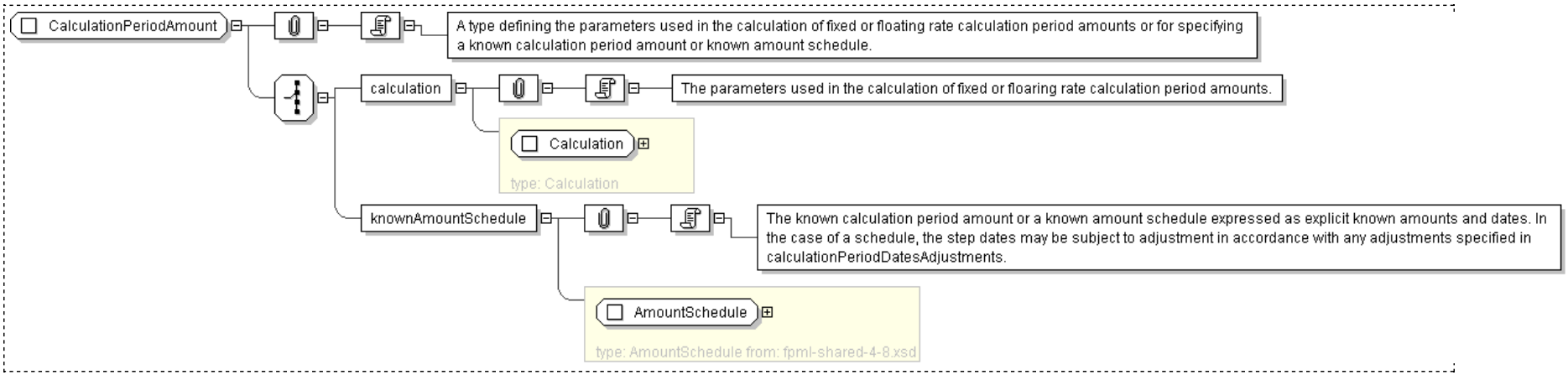
Super-types:	None
Sub-types:	None

Name	CalculationPeriodAmount
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining the parameters used in the calculation of fixed or floating rate calculation period amounts or for specifying a known calculation period amount or known amount schedule.

XML Instance Representation

```
<...>
  Start Choice [1]
    <calculation> Calculation </calculation> [1]
      'The parameters used in the calculation of fixed or floaring rate calculation period amounts.'

    <knownAmountSchedule> AmountSchedule </knownAmountSchedule> [1]
      'The known calculation period amount or a known amount schedule expressed as explicit known amounts and dates. In the case of a schedule, the
      step dates may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodAmount">
  <xsd:choice>
    <xsd:element name="calculation" type=" Calculation "/>
    <xsd:element name="knownAmountSchedule" type=" AmountSchedule "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodDates

[Table of contents]

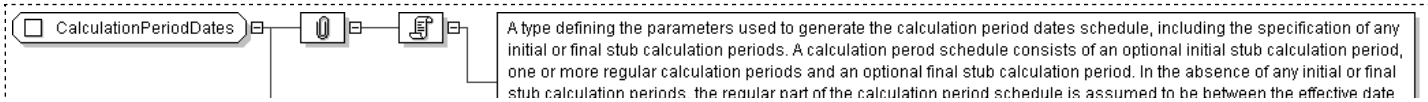
Super-types:	None
Sub-types:	None

Name	CalculationPeriodDates
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining the parameters used to generate the calculation period dates schedule, including the specification of any initial or final stub calculation periods. A calculation perod schedule consists of an optional initial stub calculation period, one or more regular calculation periods and an optional final stub calculation period. In the absence of any initial or final stub calculation periods, the regular part of the calculation period schedule is assumed to be between the effective date and the termination date. No implicit stubs are allowed, i.e. stubs must be explicitly specified using an appropriate combination of firstPeriodStateDate, firstRegularPeriodStartDate and lastRegularPeriodEndDate.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
   Start Choice [1]
      <effectiveDate> AdjustableDate </effectiveDate> [1]
      'The first day of the term of the trade. This day may be subject to adjustment in accordance with a business day convention.'

      <relativeEffectiveDate> AdjustedRelativeDateOffset </relativeEffectiveDate> [1]
      'Defines the effective date.'

   End Choice
   Start Choice [1]
      <terminationDate> AdjustableDate </terminationDate> [1]
      'The last day of the term of the trade. This day may be subject to adjustment in accordance with a business day convention.'

      <relativeTerminationDate> RelativeDateOffset </relativeTerminationDate> [1]
      'The term/maturity of the swap, express as a tenor (typically in years).'

   End Choice
   <calculationPeriodDatesAdjustments> BusinessDayAdjustments </calculationPeriodDatesAdjustments> [1]
   'The business day convention to apply to each calculation period end date if it would otherwise fall on a day that is not a business day in the
   specified financial business centers.'

   <firstPeriodStartDate> AdjustableDate </firstPeriodStartDate> [0..1]
   'The start date of the calculation period if the date falls before the effective date. It must only be specified if it is not equal to the effective
   date. This date may be subject to adjustment in accordance with a business day convention.'

   <firstRegularPeriodStartDate> xsd:date </firstRegularPeriodStartDate> [0..1]
   'The start date of the regular part of the calculation period schedule. It must only be specified if there is an initial stub calculation period. This
   day may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.'

   <firstCompoundingPeriodEndDate> xsd:date </firstCompoundingPeriodEndDate> [0..1]
   'The end date of the initial compounding period when compounding is applicable. It must only be specified when the compoundingMethod element is present
   and not equal to a value of None. This date may be subject to adjustment in accordance with any adjustments specified in
   calculationPeriodDatesAdjustments.'

   <lastRegularPeriodEndDate> xsd:date </lastRegularPeriodEndDate> [0..1]
   'The end date of the regular part of the calculation period schedule. It must only be specified if there is a final stub calculation period. This day
   may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.'

   <stubPeriodType> StubPeriodTypeEnum </stubPeriodType> [0..1]
   'Method to allocate any irregular period remaining after regular periods have been allocated between the effective and termination date.'

   <calculationPeriodFrequency> CalculationPeriodFrequency </calculationPeriodFrequency> [1]
   'The frequency at which calculation period end dates occur with the regular part of the calculation period schedule and their roll date convention.'

</...>
```

Diagram



and the termination date. No implicit stubs are allowed, i.e. stubs must be explicitly specified using an appropriate combination of firstPeriodStateDate, firstRegularPeriodStartDate and lastRegularPeriodEndDate.

effectiveDate

The first day of the term of the trade. This day may be subject to adjustment in accordance with a business day convention.

AdjustableDate

type: AdjustableDate from: fpml-shared-4-8.xsd

relativeEffectiveDate

Defines the effective date.

AdjustedRelativeDateOffset

type: AdjustedRelativeDateOffset from: fpml-shared-4-8.xsd

terminationDate

The last day of the term of the trade. This day may be subject to adjustment in accordance with a business day convention.

AdjustableDate

type: AdjustableDate from: fpml-shared-4-8.xsd

relativeTerminationDate

The term/maturity of the swap, express as a tenor (typically in years).

RelativeDateOffset

type: RelativeDateOffset from: fpml-shared-4-8.xsd

calculationPeriodDatesAdjustments

The business day convention to apply to each calculation period end date if it would otherwise fall on a day that is not a business day in the specified financial business centers.

BusinessDayAdjustments

type: BusinessDayAdjustments from: fpml-shared-4-8.xsd

firstPeriodStartDate

0..1

The start date of the calculation period if the date falls before the effective date. It must only be specified if it is not equal to the effective date. This date may be subject to adjustment in accordance with a business day convention.

AdjustableDate

type: AdjustableDate from: fpml-shared-4-8.xsd

firstRegularPeriodStartDate

0..1

The start date of the regular part of the calculation period schedule. It must only be specified if there is an initial stub calculation period. This day may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.

date

type: date from: XMLSchema.xsd

firstCompoundingPeriodEndDate

0..1

The end date of the initial compounding period when compounding is applicable. It must only be specified when the compoundingMethod element is present and not equal to a value of None. This date may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.

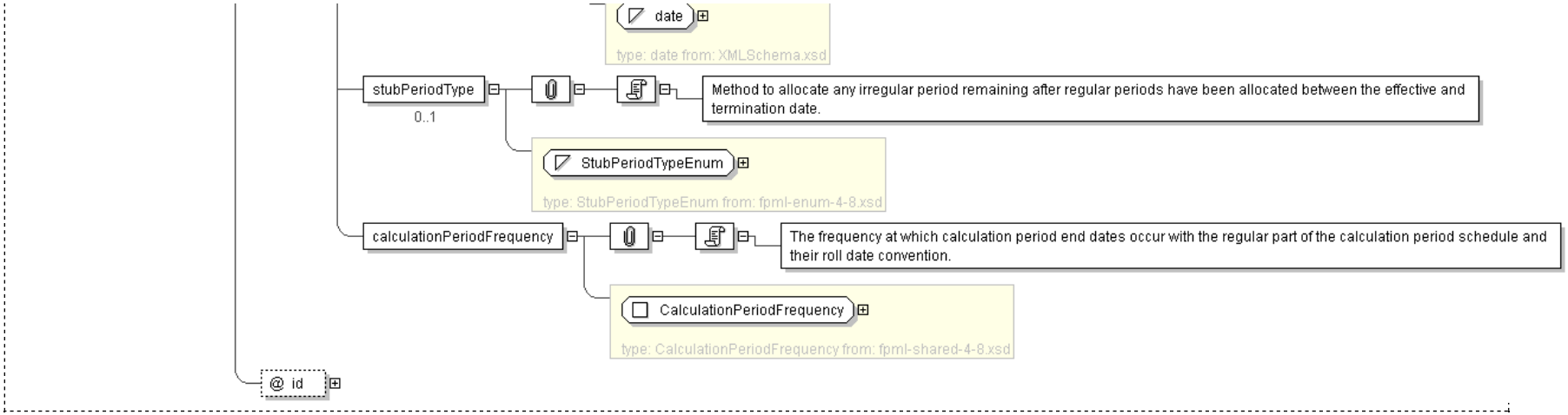
date

type: date from: XMLSchema.xsd

lastRegularPeriodEndDate

0..1

The end date of the regular part of the calculation period schedule. It must only be specified if there is a final stub calculation period. This day may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodDates">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="effectiveDate" type=" AdjustableDate "/>
         <xsd:element name="relativeEffectiveDate" type=" AdjustedRelativeDateOffset "/>
      </xsd:choice>
      <xsd:choice>
         <xsd:element name="terminationDate" type=" AdjustableDate "/>
         <xsd:element name="relativeTerminationDate" type=" RelativeDateOffset "/>
      </xsd:choice>
      <xsd:element name="calculationPeriodDatesAdjustments" type=" BusinessDayAdjustments "/>
      <xsd:element name="firstPeriodStartDate" type=" AdjustableDate " minOccurs="0"/>
      <xsd:element name="firstRegularPeriodStartDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="firstCompoundingPeriodEndDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="lastRegularPeriodEndDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="stubPeriodType" type=" StubPeriodTypeEnum " minOccurs="0"/>
      <xsd:element name="calculationPeriodFrequency" type=" CalculationPeriodFrequency "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodDatesReference

[Table of contents]

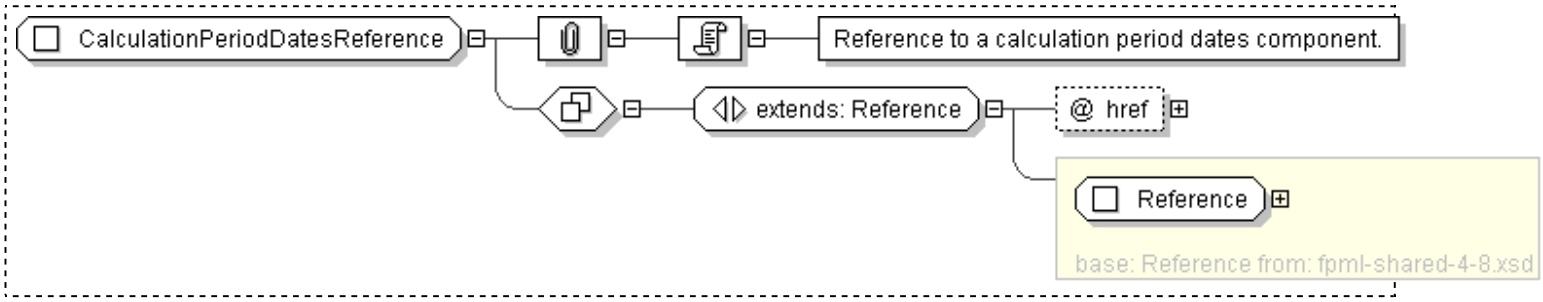
Super-types:	Reference < CalculationPeriodDatesReference (by extension)
Sub-types:	None

Name	CalculationPeriodDatesReference
Used by (from the same schema document)	Complex Type DateRelativeToCalculationPeriodDates , Complex Type NotionalStepRule , Complex Type PaymentDates , Complex Type ResetDates , Complex Type StubCalculationPeriodAmount
Abstract	no
Documentation	Reference to a calculation period dates component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodDatesReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="CalculationPeriodDates"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancelableProvision

[Table of contents]

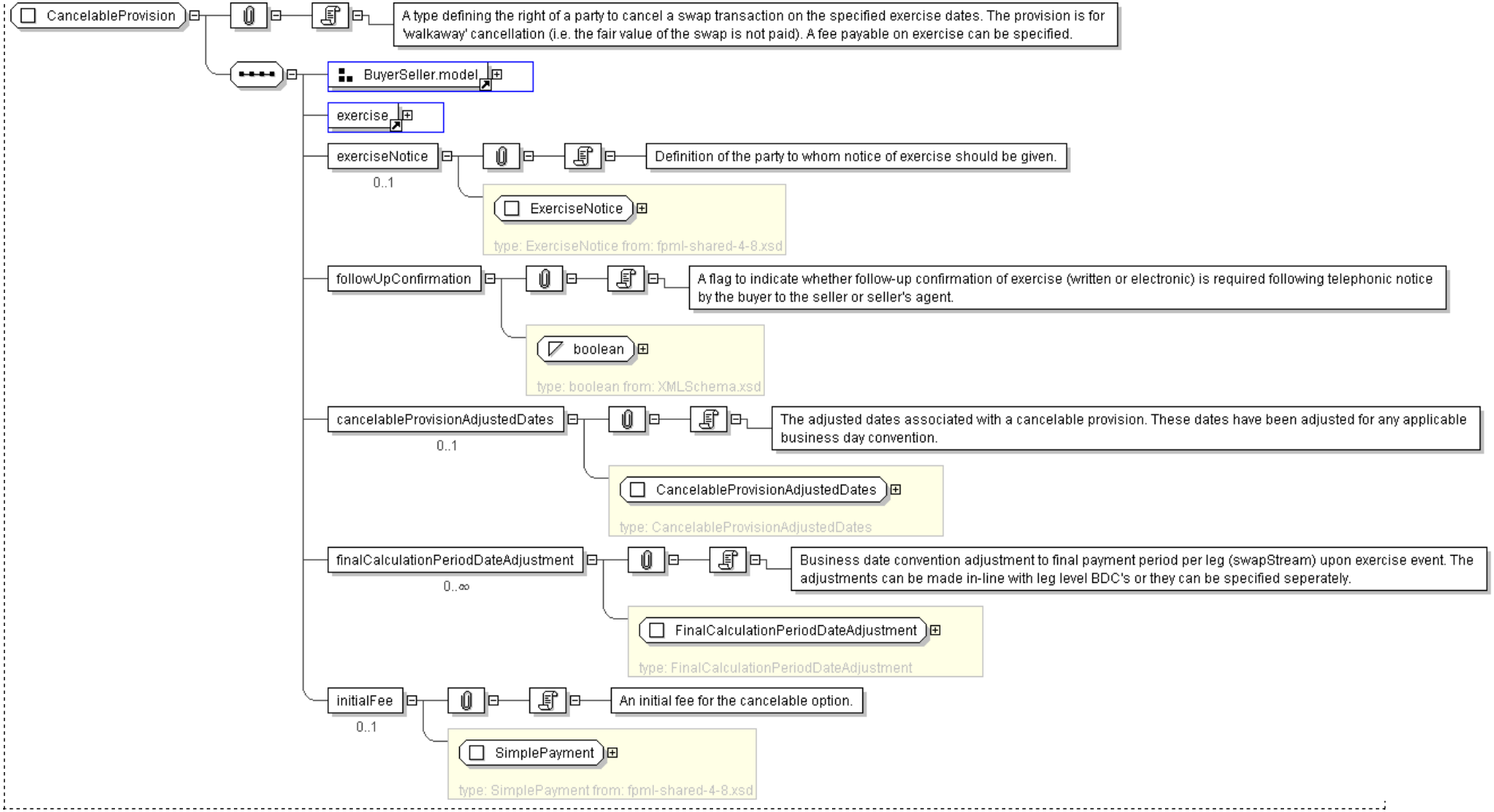
Super-types:	None
Sub-types:	None

Name	CancelableProvision
Used by (from the same schema document)	Complex Type Swap
Abstract	no
Documentation	A type defining the right of a party to cancel a swap transaction on the specified exercise dates. The provision is for 'walkaway' cancellation (i.e. the fair value of the swap is not paid). A fee payable on exercise can be specified.

XML Instance Representation

```
<...>
   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA
   definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives
   a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <exercise> ... </exercise> [1]
   <exerciseNotice> ExerciseNotice </exerciseNotice> [0..1]
   'Definition of the party to whom notice of exercise should be given.'

   <followUpConfirmation> xsd:boolean </followUpConfirmation> [1]
   'A flag to indicate whether follow-up confirmation of exercise (written or electronic) is required following telephonic notice by the buyer to the
   seller or seller\'s agent.'

   <cancelableProvisionAdjustedDates> CancelableProvisionAdjustedDates </cancelableProvisionAdjustedDates> [0..1]
   'The adjusted dates associated with a cancelable provision. These dates have been adjusted for any applicable business day convention.'

   <finalCalculationPeriodDateAdjustment> FinalCalculationPeriodDateAdjustment </finalCalculationPeriodDateAdjustment> [0..*]
   'Business date convention adjustment to final payment period per leg (swapStream) upon exercise event. The adjustments can be made in-line with
   leg level BDC\'s or they can be specified seperately.'

   <initialFee> SimplePayment </initialFee> [0..1]
   'An initial fee for the cancelable option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancelableProvision">
  <xsd:sequence>
    <xsd:group ref=" BuyerSeller.model "/>
    <xsd:element ref=" exercise "/>
    <xsd:element name="exerciseNotice" type=" ExerciseNotice " minOccurs="0"/>
    <xsd:element name="followUpConfirmation" type=" xsd:boolean "/>
    <xsd:element name="cancelableProvisionAdjustedDates" type=" CancelableProvisionAdjustedDates " minOccurs="0"/>
    <xsd:element name="finalCalculationPeriodDateAdjustment" type=" FinalCalculationPeriodDateAdjustment " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="initialFee" type=" SimplePayment " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancelableProvisionAdjustedDates

[Table of contents]

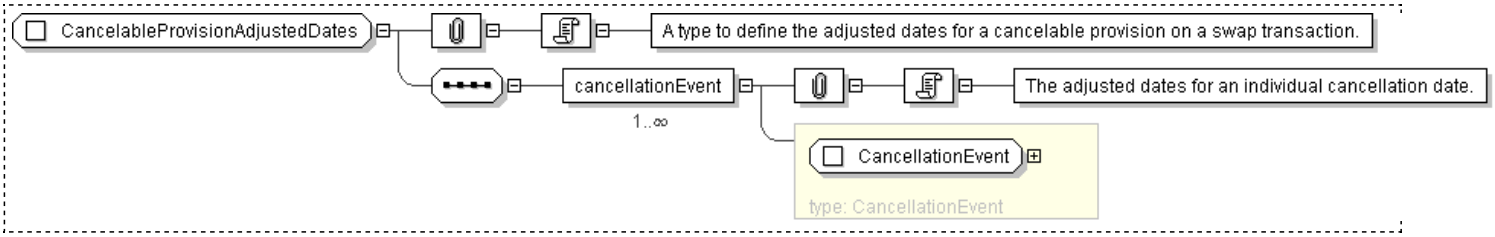
Super-types:	None
Sub-types:	None

Name	CancelableProvisionAdjustedDates
Used by (from the same schema document)	Complex Type CancelableProvision
Abstract	no
Documentation	A type to define the adjusted dates for a cancelable provision on a swap transaction.

XML Instance Representation

```
<...>
  <cancellationEvent> CancellationEvent </cancellationEvent> [1..*]
  'The adjusted dates for an individual cancellation date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancelableProvisionAdjustedDates">
  <xsd:sequence>
    <xsd:element name="cancellationEvent" type=" CancellationEvent " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancellationEvent

[Table of contents]

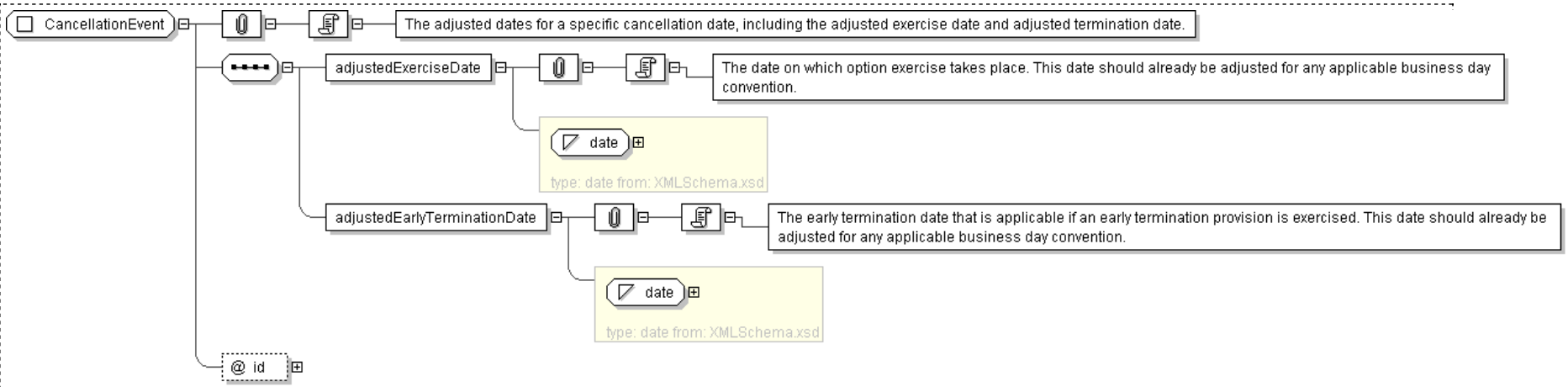
Super-types:	None
Sub-types:	None

Name	CancellationEvent
Used by (from the same schema document)	Complex Type CancelableProvisionAdjustedDates
Abstract	no
Documentation	The adjusted dates for a specific cancellation date, including the adjusted exercise date and adjusted termination date.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <adjustedExerciseDate> xsd:date </adjustedExerciseDate> [1]
   'The date on which option exercise takes place. This date should already be adjusted for any applicable business day convention.'

   <adjustedEarlyTerminationDate> xsd:date </adjustedEarlyTerminationDate> [1]
   'The early termination date that is applicable if an early termination provision is exercised. This date should already be adjusted for any
   applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancellationEvent">
  <xsd:sequence>
    <xsd:element name="adjustedExerciseDate" type=" xsd:date "/>
    <xsd:element name="adjustedEarlyTerminationDate" type=" xsd:date "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CapFloor

[Table of contents]

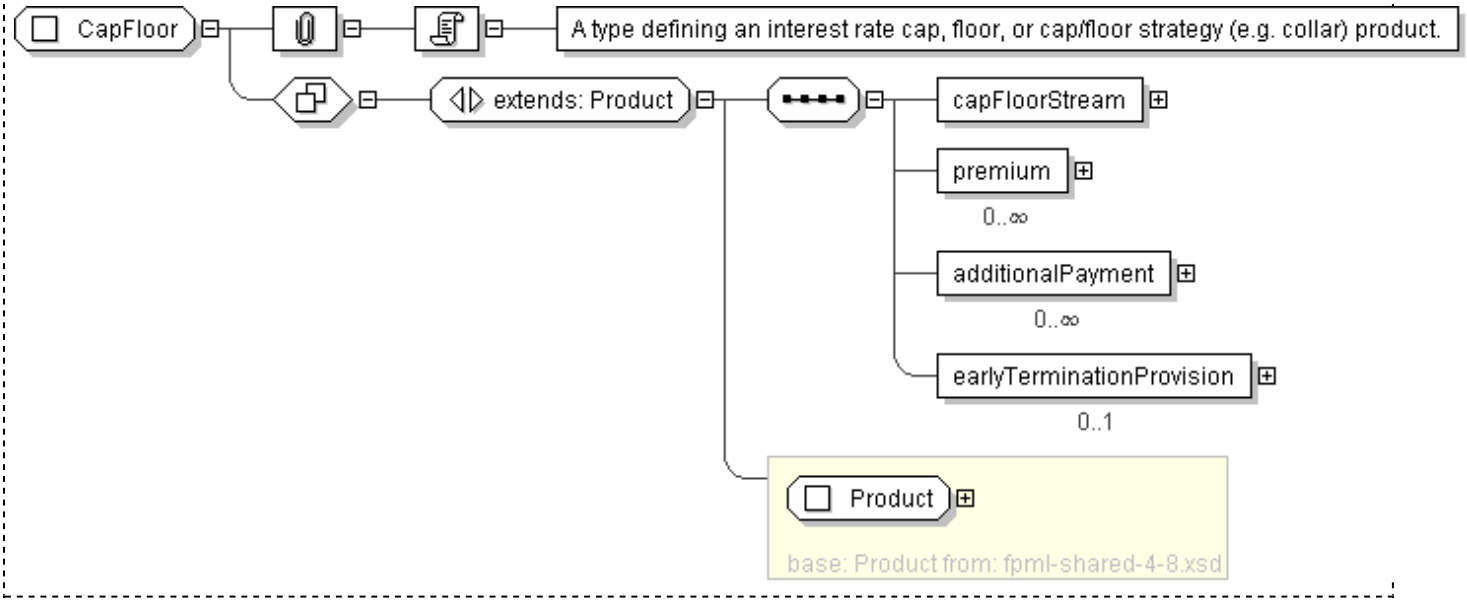
Super-types:	Product < CapFloor (by extension)
Sub-types:	None

Name	CapFloor
Used by (from the same schema document)	Element capFloor
Abstract	no
Documentation	A type defining an interest rate cap, floor, or cap/floor strategy (e.g. collar) product.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for this
   element are not strictly an enumerated list.'

   <capFloorStream> InterestRateStream </capFloorStream> [1]
   <premium> Payment </premium> [0..*]
   'The option premium amount payable by buyer to seller on the specified payment
   date.'

   <additionalPayment> Payment </additionalPayment> [0..*]
   'Additional payments between the principal parties.'

   <earlyTerminationProvision> EarlyTerminationProvision </earlyTerminationProvision>
   [0..1]
   'Parameters specifying provisions relating to the optional and mandatory early
   terminarion of a CapFloor transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CapFloor">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="capFloorStream" type=" InterestRateStream "/>
        <xsd:element name="premium" type=" Payment " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="additionalPayment" type=" Payment " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="earlyTerminationProvision" type="
        EarlyTerminationProvision " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Cashflows

[Table of contents]

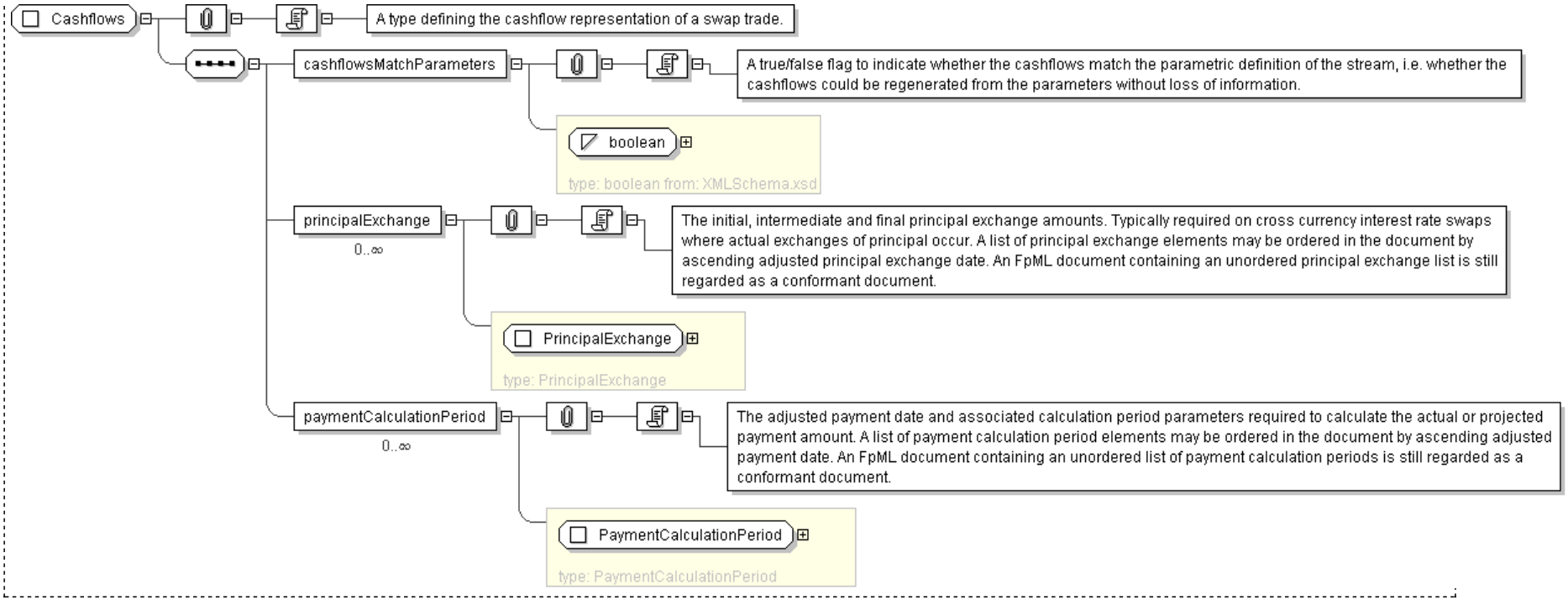
Super-types:	None
Sub-types:	None

Name	Cashflows
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining the cashflow representation of a swap trade.

XML Instance Representation

```
<...>
   <cashflowsMatchParameters> xsd:boolean </cashflowsMatchParameters> [1]
   'A true/false flag to indicate whether the cashflows match the parametric definition of the stream, i.e. whether the cashflows could be
   regenerated from the parameters without loss of information.'

   <principalExchange> PrincipalExchange </principalExchange> [0..*]
   'The initial, intermediate and final principal exchange amounts. Typically required on cross currency interest rate swaps where actual
   exchanges of principal occur. A list of principal exchange elements may be ordered in the document by ascending adjusted principal
   exchange date. An FpML document containing an unordered principal exchange list is still regarded as a conformant document.'

   <paymentCalculationPeriod> PaymentCalculationPeriod </paymentCalculationPeriod> [0..*]
   'The adjusted payment date and associated calculation period parameters required to calculate the actual or projected payment amount. A
   list of payment calculation period elements may be ordered in the document by ascending adjusted payment date. An FpML document containing
   an unordered list of payment calculation periods is still regarded as a conformant document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Cashflows">
  <xsd:sequence>
    <xsd:element name="cashflowsMatchParameters" type=" xsd:boolean "/>
    <xsd:element name="principalExchange" type=" PrincipalExchange " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="paymentCalculationPeriod" type=" PaymentCalculationPeriod " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashPriceMethod

[Table of contents]

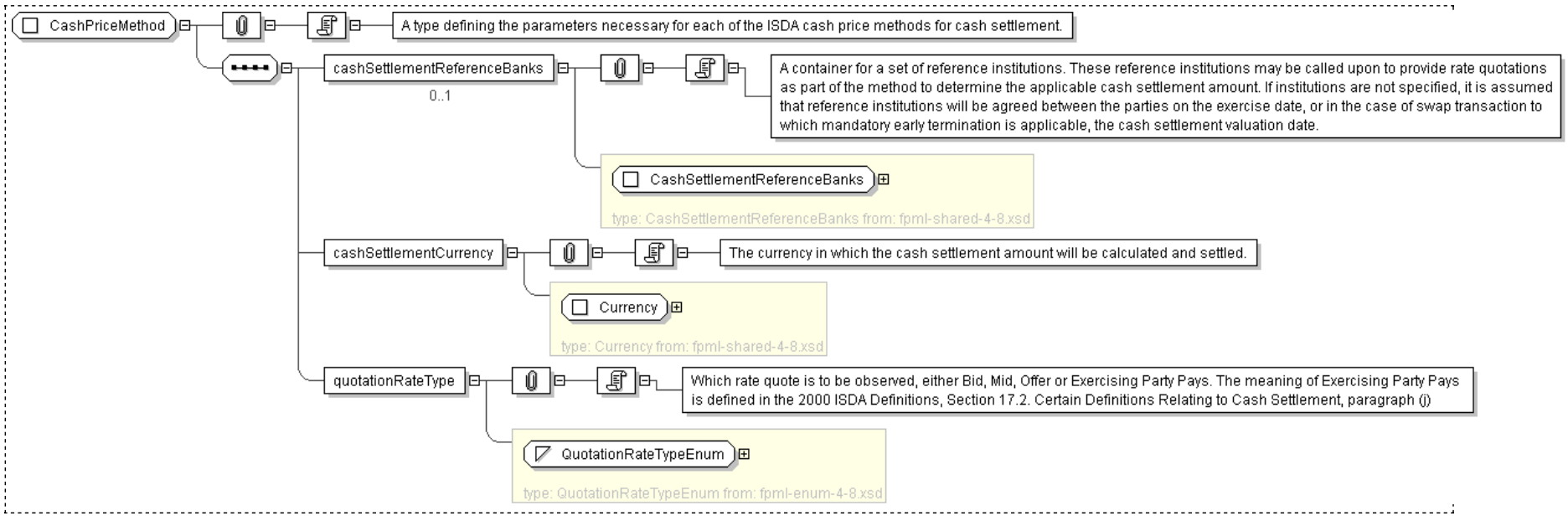
Super-types:	None
Sub-types:	None

Name	CashPriceMethod
Used by (from the same schema document)	Complex Type CashSettlement , Complex Type CashSettlement
Abstract	no
Documentation	A type defining the parameters necessary for each of the ISDA cash price methods for cash settlement.

XML Instance Representation

```
<...>
  <cashSettlementReferenceBanks> CashSettlementReferenceBanks </cashSettlementReferenceBanks> [0..1]
   'A container for a set of reference institutions. These reference institutions may be called upon to provide rate quotations as part of the method
   to determine the applicable cash settlement amount. If institutions are not specified, it is assumed that reference institutions will be agreed
   between the parties on the exercise date, or in the case of swap transaction to which mandatory early termination is applicable, the cash
   settlement valuation date.'

  <cashSettlementCurrency> Currency </cashSettlementCurrency> [1]
   'The currency in which the cash settlement amount will be calculated and settled.'

  <quotationRateType> QuotationRateTypeEnum </quotationRateType> [1]
   'Which rate quote is to be observed, either Bid, Mid, Offer or Exercising Party Pays. The meaning of Exercising Party Pays is defined in the 2000
   ISDA Definitions, Section 17.2. Certain Definitions Relating to Cash Settlement, paragraph (j)'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashPriceMethod">
  <xsd:sequence>
    <xsd:element name="cashSettlementReferenceBanks" type=" CashSettlementReferenceBanks " minOccurs="0"/>
```

```
      <xsd:element name="cashSettlementCurrency" type=" Currency "/>
      <xsd:element name="quotationRateType" type=" QuotationRateTypeEnum "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashSettlement

[Table of contents]

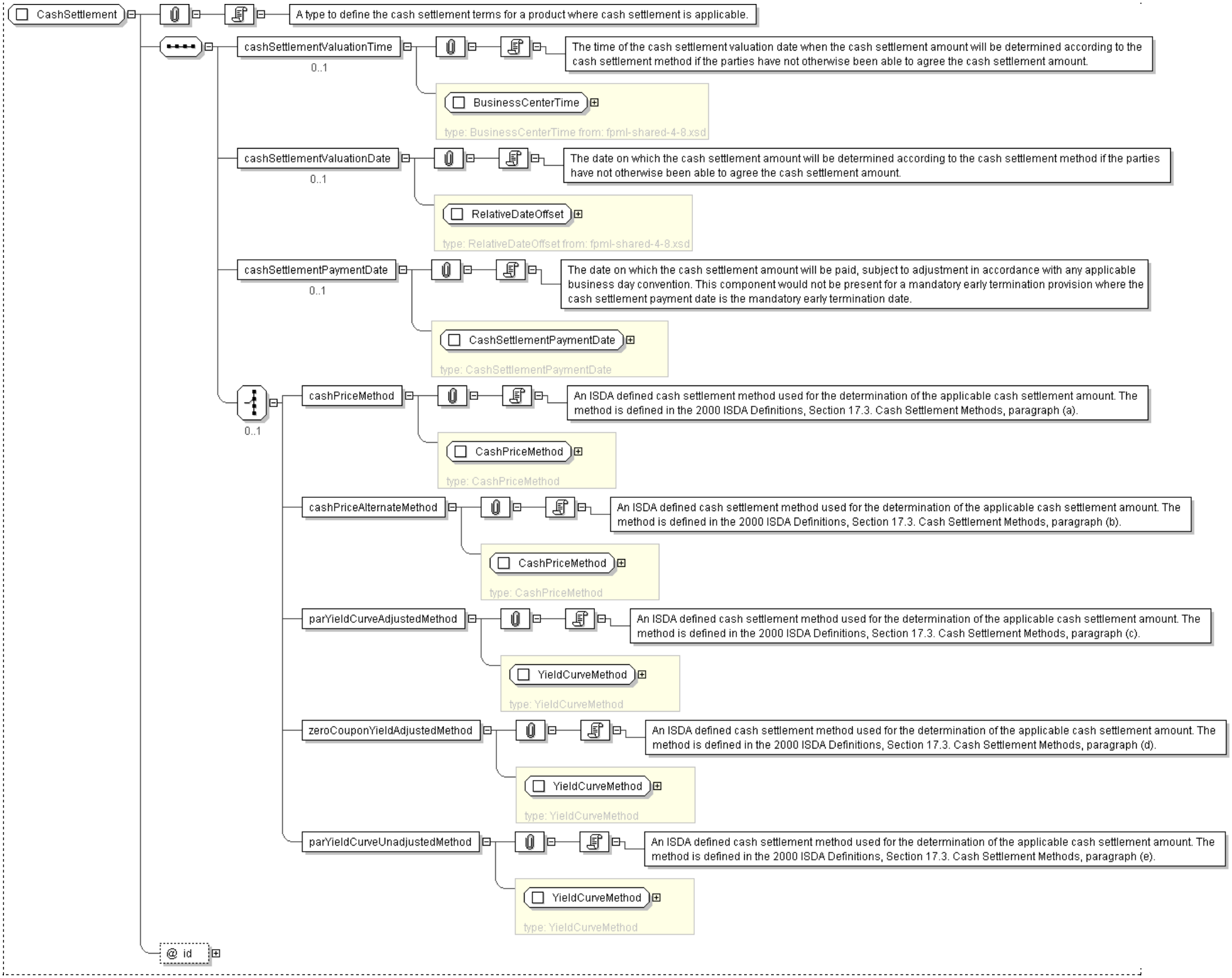
Super-types:	None
Sub-types:	None

Name	CashSettlement
Used by (from the same schema document)	Complex Type MandatoryEarlyTermination , Complex Type OptionalEarlyTermination , Complex Type Swaption
Abstract	no
Documentation	A type to define the cash settlement terms for a product where cash settlement is applicable.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <cashSettlementValuationTime> BusinessCenterTime </cashSettlementValuationTime> [0..1]
    'The time of the cash settlement valuation date when the cash settlement amount will be determined according to the cash settlement method if the
    parties have not otherwise been able to agree the cash settlement amount.'

   <cashSettlementValuationDate> RelativeDateOffset </cashSettlementValuationDate> [0..1]
    'The date on which the cash settlement amount will be determined according to the cash settlement method if the parties have not otherwise been able to
    agree the cash settlement amount.'

   <cashSettlementPaymentDate> CashSettlementPaymentDate </cashSettlementPaymentDate> [0..1]
    'The date on which the cash settlement amount will be paid, subject to adjustment in accordance with any applicable business day convention. This
    component would not be present for a mandatory early termination provision where the cash settlement payment date is the mandatory early termination
    date.'

   Start Choice [0..1]
      <cashPriceMethod> CashPriceMethod </cashPriceMethod> [1]
       'An ISDA defined cash settlement method used for the determination of the applicable cash settlement amount. The method is defined in the 2000 ISDA
       Definitions, Section 17.3. Cash Settlement Methods, paragraph (a).'

      <cashPriceAlternateMethod> CashPriceMethod </cashPriceAlternateMethod> [1]
       'An ISDA defined cash settlement method used for the determination of the applicable cash settlement amount. The method is defined in the 2000 ISDA
       Definitions, Section 17.3. Cash Settlement Methods, paragraph (b).'

      <parYieldCurveAdjustedMethod> YieldCurveMethod </parYieldCurveAdjustedMethod> [1]
       'An ISDA defined cash settlement method used for the determination of the applicable cash settlement amount. The method is defined in the 2000 ISDA
       Definitions, Section 17.3. Cash Settlement Methods, paragraph (c).'

      <zeroCouponYieldAdjustedMethod> YieldCurveMethod </zeroCouponYieldAdjustedMethod> [1]
       'An ISDA defined cash settlement method used for the determination of the applicable cash settlement amount. The method is defined in the 2000 ISDA
       Definitions, Section 17.3. Cash Settlement Methods, paragraph (d).'

      <parYieldCurveUnadjustedMethod> YieldCurveMethod </parYieldCurveUnadjustedMethod> [1]
       'An ISDA defined cash settlement method used for the determination of the applicable cash settlement amount. The method is defined in the 2000 ISDA
       Definitions, Section 17.3. Cash Settlement Methods, paragraph (e).'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashSettlement">
   <xsd:sequence>
      <xsd:element name="cashSettlementValuationTime" type=" BusinessCenterTime " minOccurs="0"/>
      <xsd:element name="cashSettlementValuationDate" type=" RelativeDateOffset " minOccurs="0"/>
      <xsd:element name="cashSettlementPaymentDate" type=" CashSettlementPaymentDate " minOccurs="0"/>
      <xsd:choice minOccurs="0">
         <xsd:element name="cashPriceMethod" type=" CashPriceMethod "/>
         <xsd:element name="cashPriceAlternateMethod" type=" CashPriceMethod "/>
         <xsd:element name="parYieldCurveAdjustedMethod" type=" YieldCurveMethod "/>
         <xsd:element name="zeroCouponYieldAdjustedMethod" type=" YieldCurveMethod "/>
         <xsd:element name="parYieldCurveUnadjustedMethod" type=" YieldCurveMethod "/>
      </xsd:choice>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashSettlementPaymentDate

[Table of contents]

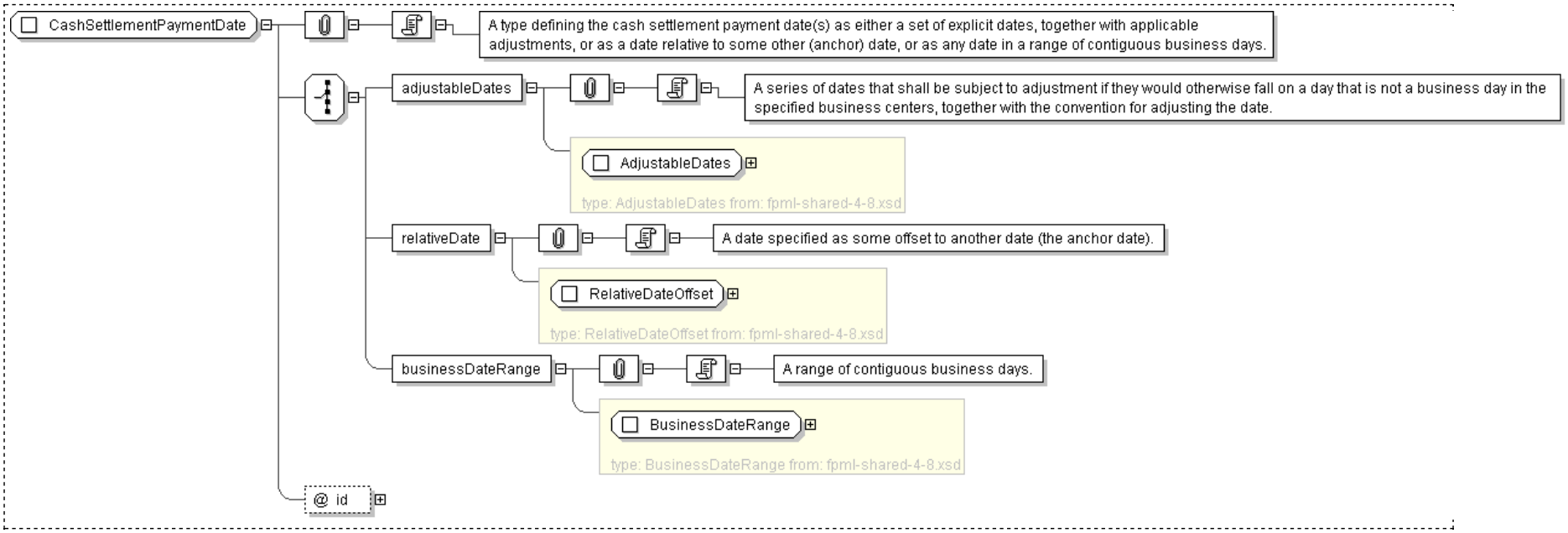
Super-types:	None
Sub-types:	None

Name	CashSettlementPaymentDate
Used by (from the same schema document)	Complex Type CashSettlement
Abstract	no
Documentation	A type defining the cash settlement payment date(s) as either a set of explicit dates, together with applicable adjustments, or as a date relative to some other (anchor) date, or as any date in a range of contiguous business days.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <adjustableDates> AdjustableDates </adjustableDates> [1]
      'A series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified
      business centers, together with the convention for adjusting the date.'

      <relativeDate> RelativeDateOffset </relativeDate> [1]
      'A date specified as some offset to another date (the anchor date).'

      <businessDateRange> BusinessDateRange </businessDateRange> [1]
      'A range of contiguous business days.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashSettlementPaymentDate">
  <xsd:choice>
    <xsd:element name="adjustableDates" type=" AdjustableDates "/>
    <xsd:element name="relativeDate" type=" RelativeDateOffset "/>
    <xsd:element name="businessDateRange" type=" BusinessDateRange "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateRelativeToCalculationPeriodDates

[Table of contents]

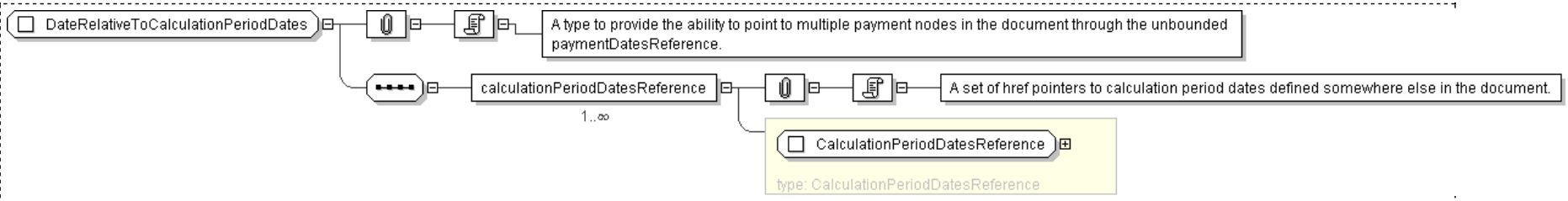
Super-types:	None
Sub-types:	None

Name	DateRelativeToCalculationPeriodDates
Used by (from the same schema document)	Complex Type FxFixingDate
Abstract	no
Documentation	A type to provide the ability to point to multiple payment nodes in the document through the unbounded paymentDatesReference.

XML Instance Representation

```
<...>
  <calculationPeriodDatesReference> CalculationPeriodDatesReference </calculationPeriodDatesReference> [1..*]
   'A set of href pointers to calculation period dates defined somewhere else in the document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateRelativeToCalculationPeriodDates">
  <xsd:sequence>
    <xsd:element name="calculationPeriodDatesReference" type=" CalculationPeriodDatesReference " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateRelativeToPaymentDates

[Table of contents]

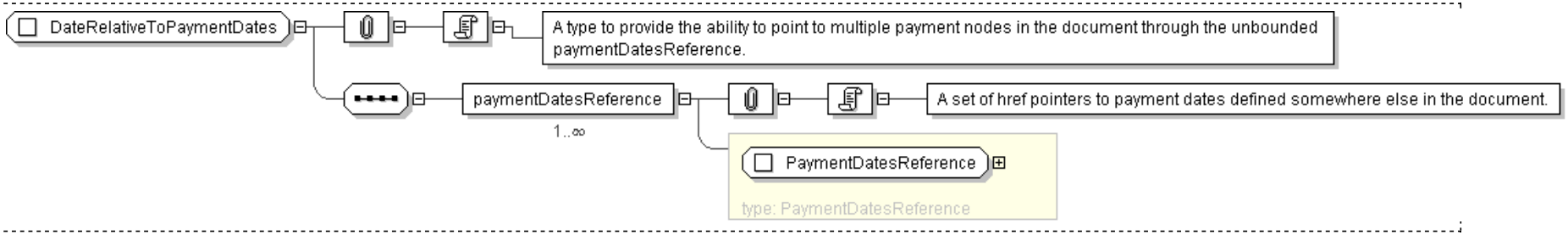
Super-types:	None
Sub-types:	None

Name	DateRelativeToPaymentDates
Used by (from the same schema document)	Complex Type FxFixingDate
Abstract	no
Documentation	A type to provide the ability to point to multiple payment nodes in the document through the unbounded paymentDatesReference.

XML Instance Representation

```
<...>
  <paymentDatesReference> PaymentDatesReference </paymentDatesReference> [1..*]
  'A set of href pointers to payment dates defined somewhere else in the document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateRelativeToPaymentDates">
  <xsd:sequence>
    <xsd:element name="paymentDatesReference" type=" PaymentDatesReference " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Discounting

[Table of contents]

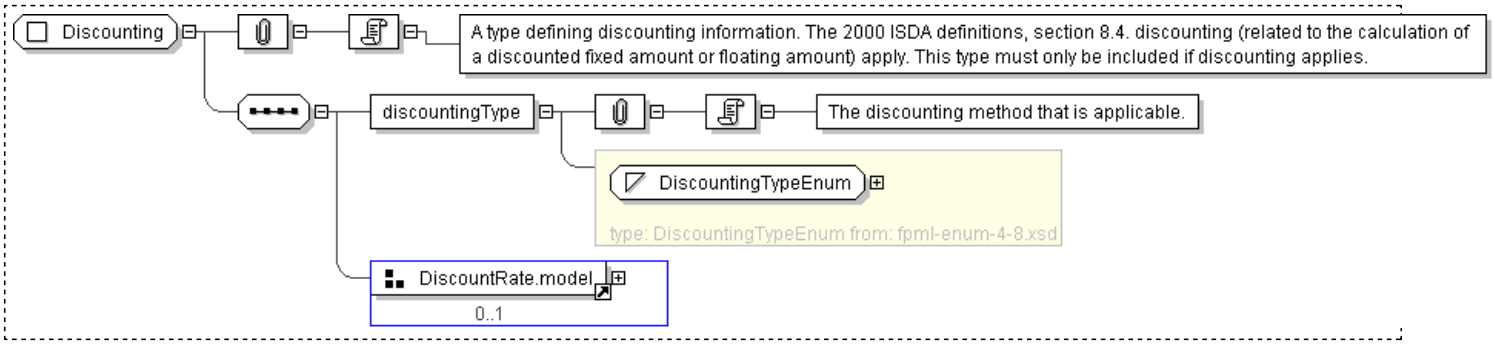
Super-types:	None
Sub-types:	None

Name	Discounting
Used by (from the same schema document)	Complex Type Calculation
Abstract	no
Documentation	A type defining discounting information. The 2000 ISDA definitions, section 8.4. discounting (related to the calculation of a discounted fixed amount or floating amount) apply. This type must only be included if discounting applies.

XML Instance Representation

```
<...>
   <discountingType> DiscountingTypeEnum </discountingType> [1]
    'The discounting method that is applicable.'

   Start Group: DiscountRate.model [0..1]
      <discountRate> xsd:decimal </discountRate> [1]
       'A discount rate, expressed as a decimal, to be used in the calculation of a discounted amount. A
       discount amount of 5% would be represented as 0.05.'

      <discountRateDayCountFraction> DayCountFraction </discountRateDayCountFraction> [0..1]
       'A discount day count fraction to be used in the calculation of a discounted amount.'

   End Group: DiscountRate.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Discounting">
   <xsd:sequence>
      <xsd:element name="discountingType" type=" DiscountingTypeEnum "/>
      <xsd:group ref=" DiscountRate.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EarlyTerminationEvent

[Table of contents]

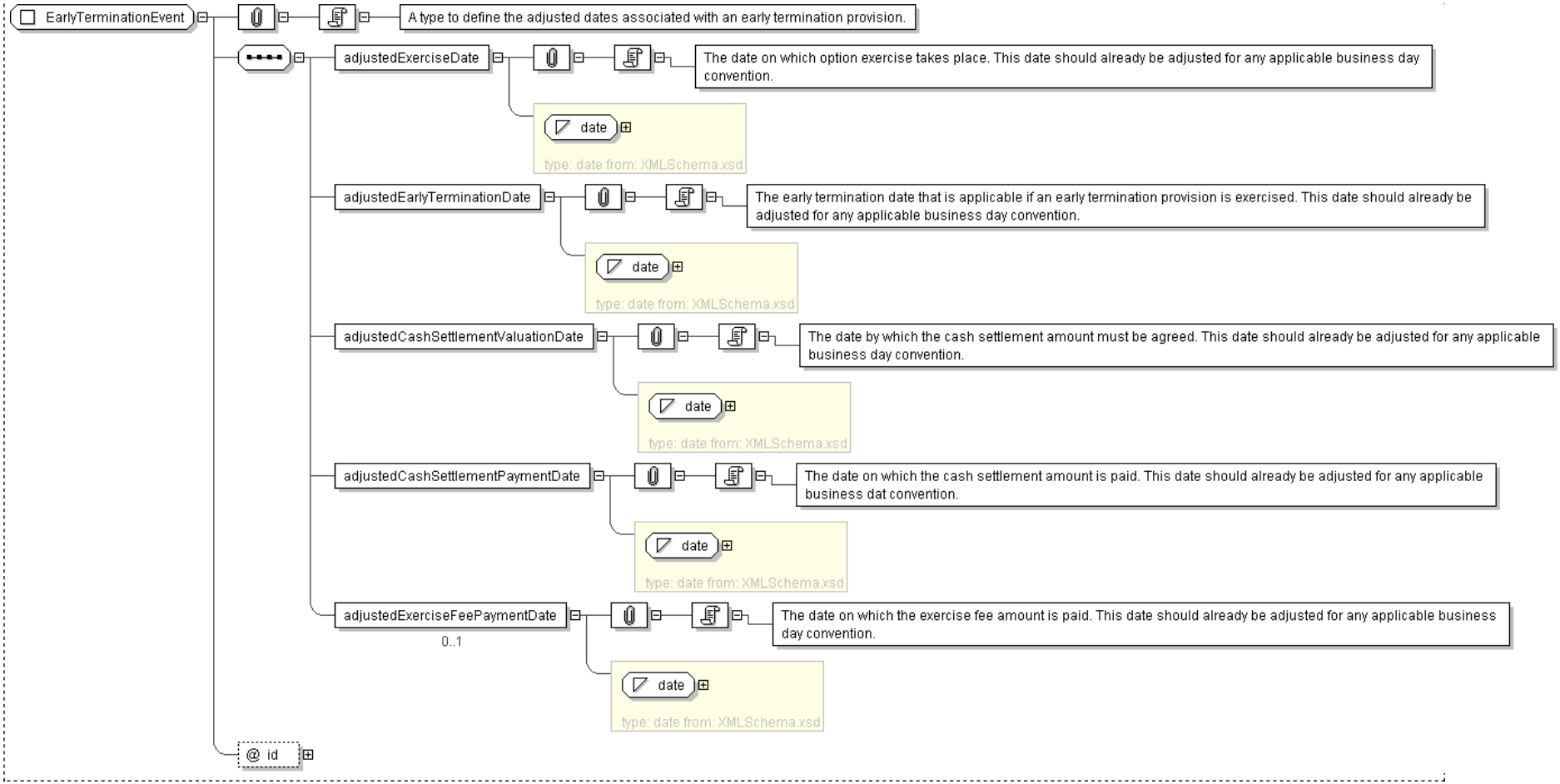
Super-types:	None
Sub-types:	None

Name	EarlyTerminationEvent
Used by (from the same schema document)	Complex Type OptionalEarlyTerminationAdjustedDates
Abstract	no
Documentation	A type to define the adjusted dates associated with an early termination provision.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <adjustedExerciseDate> xsd:date </adjustedExerciseDate> [1]
   'The date on which option exercise takes place. This date should already be adjusted for any applicable business day convention.'

   <adjustedEarlyTerminationDate> xsd:date </adjustedEarlyTerminationDate> [1]
   'The early termination date that is applicable if an early termination provision is exercised. This date should already be adjusted for any applicable
   business day convention.'

   <adjustedCashSettlementValuationDate> xsd:date </adjustedCashSettlementValuationDate> [1]
   'The date by which the cash settlement amount must be agreed. This date should already be adjusted for any applicable business day convention.'

   <adjustedCashSettlementPaymentDate> xsd:date </adjustedCashSettlementPaymentDate> [1]
   'The date on which the cash settlement amount is paid. This date should already be adjusted for any applicable business dat convention.'

   <adjustedExerciseFeePaymentDate> xsd:date </adjustedExerciseFeePaymentDate> [0..1]
   'The date on which the exercise fee amount is paid. This date should already be adjusted for any applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EarlyTerminationEvent">
  <xsd:sequence>
    <xsd:element name="adjustedExerciseDate" type=" xsd:date "/>
    <xsd:element name="adjustedEarlyTerminationDate" type=" xsd:date "/>
    <xsd:element name="adjustedCashSettlementValuationDate" type=" xsd:date "/>
    <xsd:element name="adjustedCashSettlementPaymentDate" type=" xsd:date "/>
    <xsd:element name="adjustedExerciseFeePaymentDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EarlyTerminationProvision

[Table of contents]

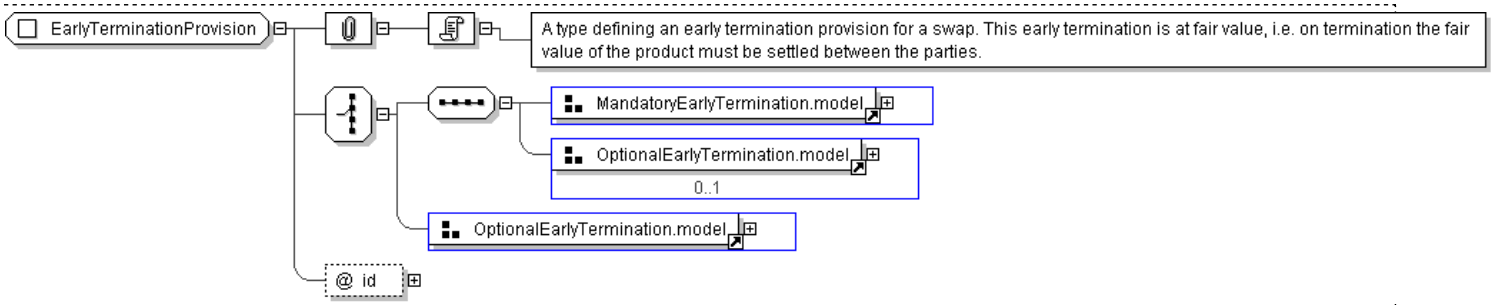
Super-types:	None
Sub-types:	None

Name	EarlyTerminationProvision
Used by (from the same schema document)	Complex Type CapFloor , Complex Type Swap
Abstract	no
Documentation	A type defining an early termination provision for a swap. This early termination is at fair value, i.e. on termination the fair value of the product must be settled between the parties.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
  Start Choice [1]
    Start Choice [1]
      <mandatoryEarlyTermination> MandatoryEarlyTermination </mandatoryEarlyTermination> [1]
      'A mandatory early termination provision to terminate the swap at fair value.'

      <mandatoryEarlyTerminationDateTenor> Period </mandatoryEarlyTerminationDateTenor> [1]
      'Period after trade date of the mandatory early termination date.'

      <mandatoryEarlyTermination> MandatoryEarlyTermination </mandatoryEarlyTermination> [0..1]
      'A mandatory early termination provision to terminate the swap at fair value.'

    End Choice
    Start Group: OptionalEarlyTermination.model [0..1]
      Start Choice [1]
        <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [1]
        'An option for either or both parties to terminate the swap at fair value.'

        <optionalEarlyTerminationParameters> ExercisePeriod </optionalEarlyTerminationParameters> [1]
        'Definition of the first early termination date and the frequency of the termination dates subsequent to
        that. American exercise is defined by having a frequency of one day.'

        <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [0..1]
        'An option for either or both parties to terminate the swap at fair value.'

      End Choice
    End Group: OptionalEarlyTermination.model
    Start Choice [1]
      <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [1]
      'An option for either or both parties to terminate the swap at fair value.'

      <optionalEarlyTerminationParameters> ExercisePeriod </optionalEarlyTerminationParameters> [1]
      'Definition of the first early termination date and the frequency of the termination dates subsequent to that.
      American exercise is defined by having a frequency of one day.'

      <optionalEarlyTermination> OptionalEarlyTermination </optionalEarlyTermination> [0..1]
      'An option for either or both parties to terminate the swap at fair value.'

    End Choice
  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EarlyTerminationProvision">
  <xsd:choice>
    <xsd:sequence>
      <xsd:group ref=" MandatoryEarlyTermination.model "/>
      <xsd:group ref=" OptionalEarlyTermination.model " minOccurs="0"/>
    </xsd:sequence>
    <xsd:group ref=" OptionalEarlyTermination.model "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExerciseEvent

[Table of contents]

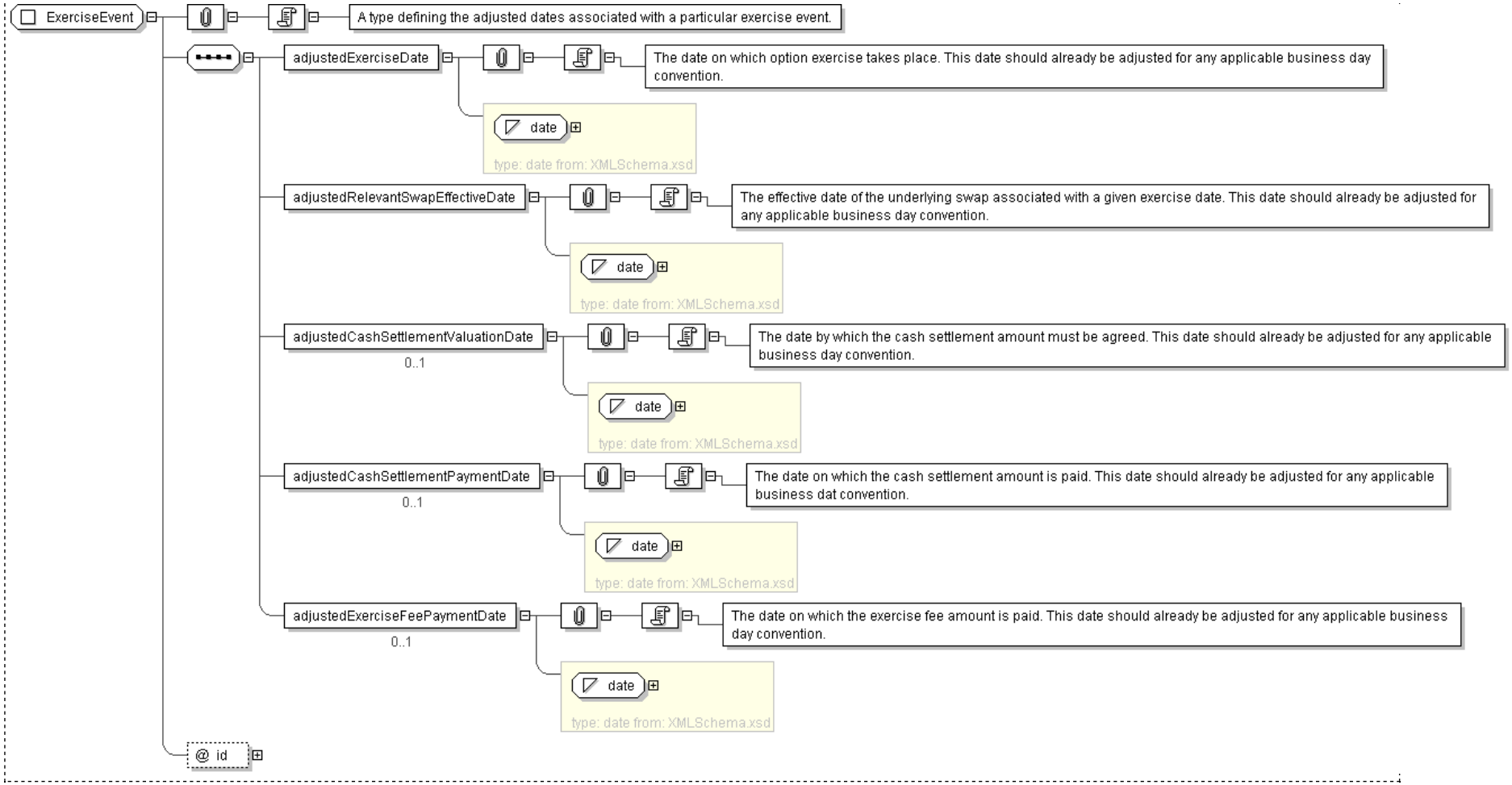
Super-types:	None
Sub-types:	None

Name	ExerciseEvent
Used by (from the same schema document)	Complex Type SwaptionAdjustedDates
Abstract	no
Documentation	A type defining the adjusted dates associated with a particular exercise event.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <adjustedExerciseDate> xsd:date </adjustedExerciseDate> [1]
    'The date on which option exercise takes place. This date should already be adjusted for any applicable business day convention.'

   <adjustedRelevantSwapEffectiveDate> xsd:date </adjustedRelevantSwapEffectiveDate> [1]
    'The effective date of the underlying swap associated with a given exercise date. This date should already be adjusted for any applicable business
    day convention.'

   <adjustedCashSettlementValuationDate> xsd:date </adjustedCashSettlementValuationDate> [0..1]
    'The date by which the cash settlement amount must be agreed. This date should already be adjusted for any applicable business day convention.'

   <adjustedCashSettlementPaymentDate> xsd:date </adjustedCashSettlementPaymentDate> [0..1]
    'The date on which the cash settlement amount is paid. This date should already be adjusted for any applicable business dat convention.'

   <adjustedExerciseFeePaymentDate> xsd:date </adjustedExerciseFeePaymentDate> [0..1]
    'The date on which the exercise fee amount is paid. This date should already be adjusted for any applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExerciseEvent">
  <xsd:sequence>
    <xsd:element name="adjustedExerciseDate" type=" xsd:date "/>
    <xsd:element name="adjustedRelevantSwapEffectiveDate" type=" xsd:date "/>
    <xsd:element name="adjustedCashSettlementValuationDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="adjustedCashSettlementPaymentDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="adjustedExerciseFeePaymentDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExercisePeriod

[Table of contents]

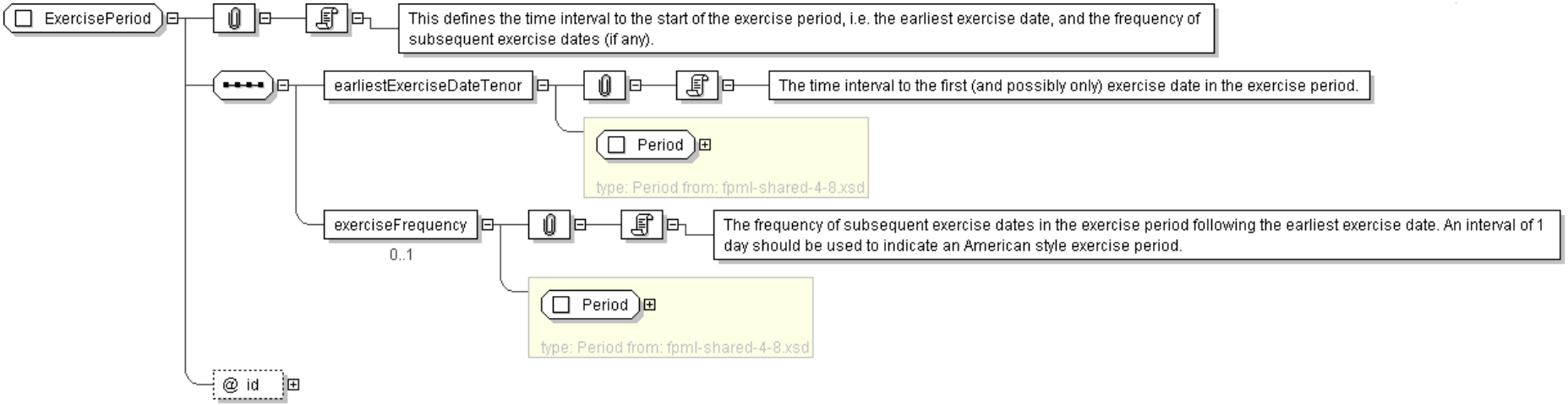
Super-types:	None
Sub-types:	None

Name	ExercisePeriod
Used by (from the same schema document)	Model Group OptionalEarlyTermination.model
Abstract	no
Documentation	This defines the time interval to the start of the exercise period, i.e. the earliest exercise date, and the frequency of subsequent exercise dates (if any).

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <earliestExerciseDateTenor> Period </earliestExerciseDateTenor> [1]
    'The time interval to the first (and possibly only) exercise date in the exercise period.'

   <exerciseFrequency> Period </exerciseFrequency> [0..1]
    'The frequency of subsequent exercise dates in the exercise period following the earliest exercise date. An interval of 1 day should be
    used to indicate an American style exercise period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExercisePeriod">
  <xsd:sequence>
    <xsd:element name="earliestExerciseDateTenor" type=" Period "/>
    <xsd:element name="exerciseFrequency" type=" Period " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExtendibleProvision

[Table of contents]

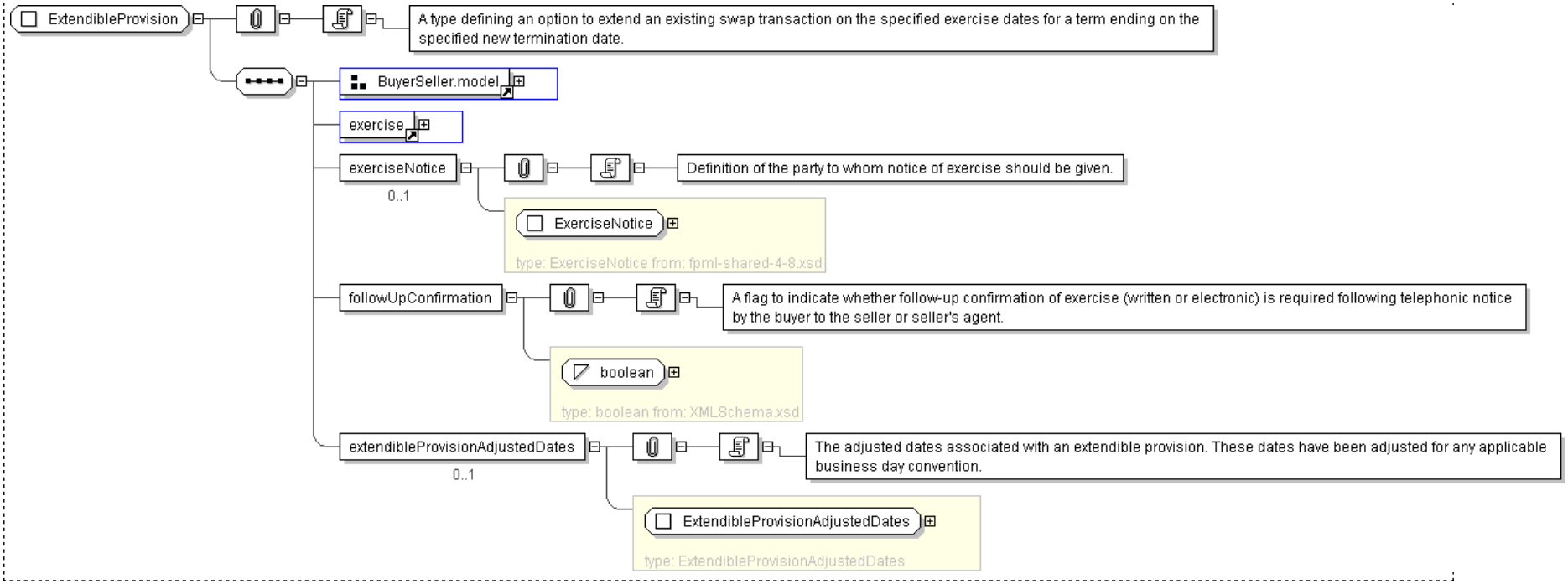
Super-types:	None
Sub-types:	None

Name	ExtendibleProvision
Used by (from the same schema document)	Complex Type Swap
Abstract	no
Documentation	A type defining an option to extend an existing swap transaction on the specified exercise dates for a term ending on the specified new termination date.

XML Instance Representation

```
<...>
  <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
  'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA
  definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

  <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
  'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives
  a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

  <exercise> ... </exercise> [1]
  <exerciseNotice> ExerciseNotice </exerciseNotice> [0..1]
  'Definition of the party to whom notice of exercise should be given.'

  <followUpConfirmation> xsd:boolean </followUpConfirmation> [1]
  'A flag to indicate whether follow-up confirmation of exercise (written or electronic) is required following telephonic notice by the buyer to
  the seller or seller\'s agent.'

  <extendibleProvisionAdjustedDates> ExtendibleProvisionAdjustedDates </extendibleProvisionAdjustedDates> [0..1]
  'The adjusted dates associated with an extendible provision. These dates have been adjusted for any applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExtendibleProvision">
  <xsd:sequence>
    <xsd:group ref=" BuyerSeller.model "/>
    <xsd:element ref=" exercise "/>
    <xsd:element name="exerciseNotice" type=" ExerciseNotice " minOccurs="0"/>
    <xsd:element name="followUpConfirmation" type=" xsd:boolean "/>
    <xsd:element name="extendibleProvisionAdjustedDates" type=" ExtendibleProvisionAdjustedDates " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExtendibleProvisionAdjustedDates

[Table of contents]

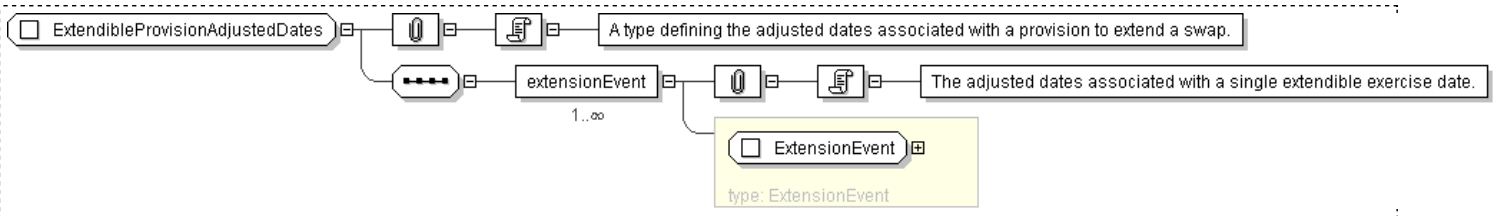
Super-types:	None
Sub-types:	None

Name	ExtendibleProvisionAdjustedDates
Used by (from the same schema document)	Complex Type ExtendibleProvision
Abstract	no
Documentation	A type defining the adjusted dates associated with a provision to extend a swap.

XML Instance Representation

```
<...>
  <extensionEvent> ExtensionEvent </extensionEvent> [1..*]
  'The adjusted dates associated with a single extendible exercise date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExtendibleProvisionAdjustedDates">
  <xsd:sequence>
    <xsd:element name="extensionEvent" type=" ExtensionEvent " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExtensionEvent

[Table of contents]

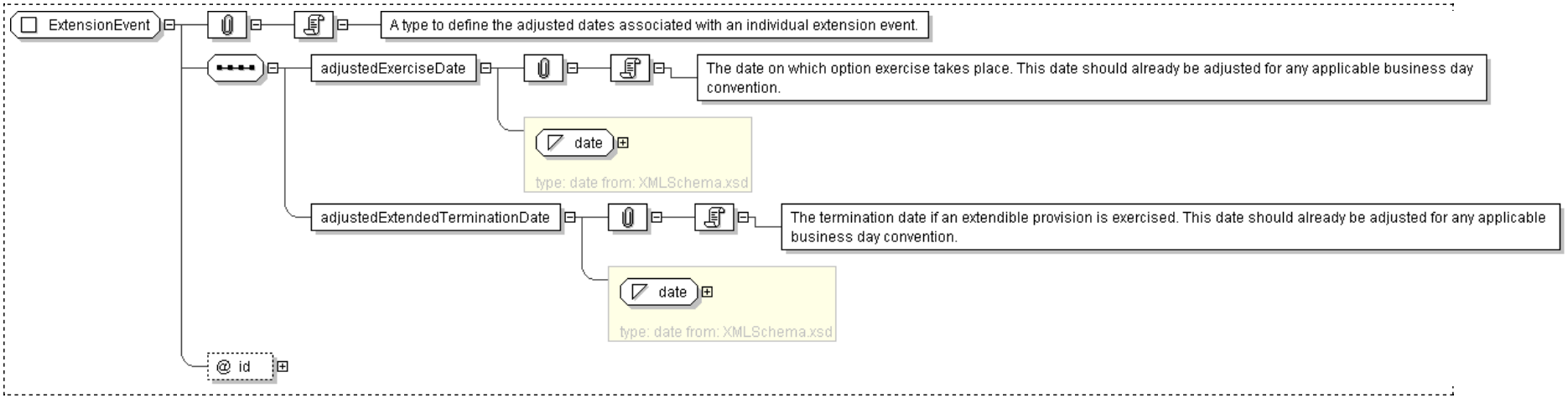
Super-types:	None
Sub-types:	None

Name	ExtensionEvent
Used by (from the same schema document)	Complex Type ExtendibleProvisionAdjustedDates
Abstract	no
Documentation	A type to define the adjusted dates associated with an individual extension event.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <adjustedExerciseDate> xsd:date </adjustedExerciseDate> [1]
    'The date on which option exercise takes place. This date should already be adjusted for any applicable business day convention.'

   <adjustedExtendedTerminationDate> xsd:date </adjustedExtendedTerminationDate> [1]
    'The termination date if an extendible provision is exercised. This date should already be adjusted for any applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExtensionEvent">
  <xsd:sequence>
    <xsd:element name="adjustedExerciseDate" type=" xsd:date "/>
    <xsd:element name="adjustedExtendedTerminationDate" type=" xsd:date "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FallbackReferencePrice

[Table of contents]

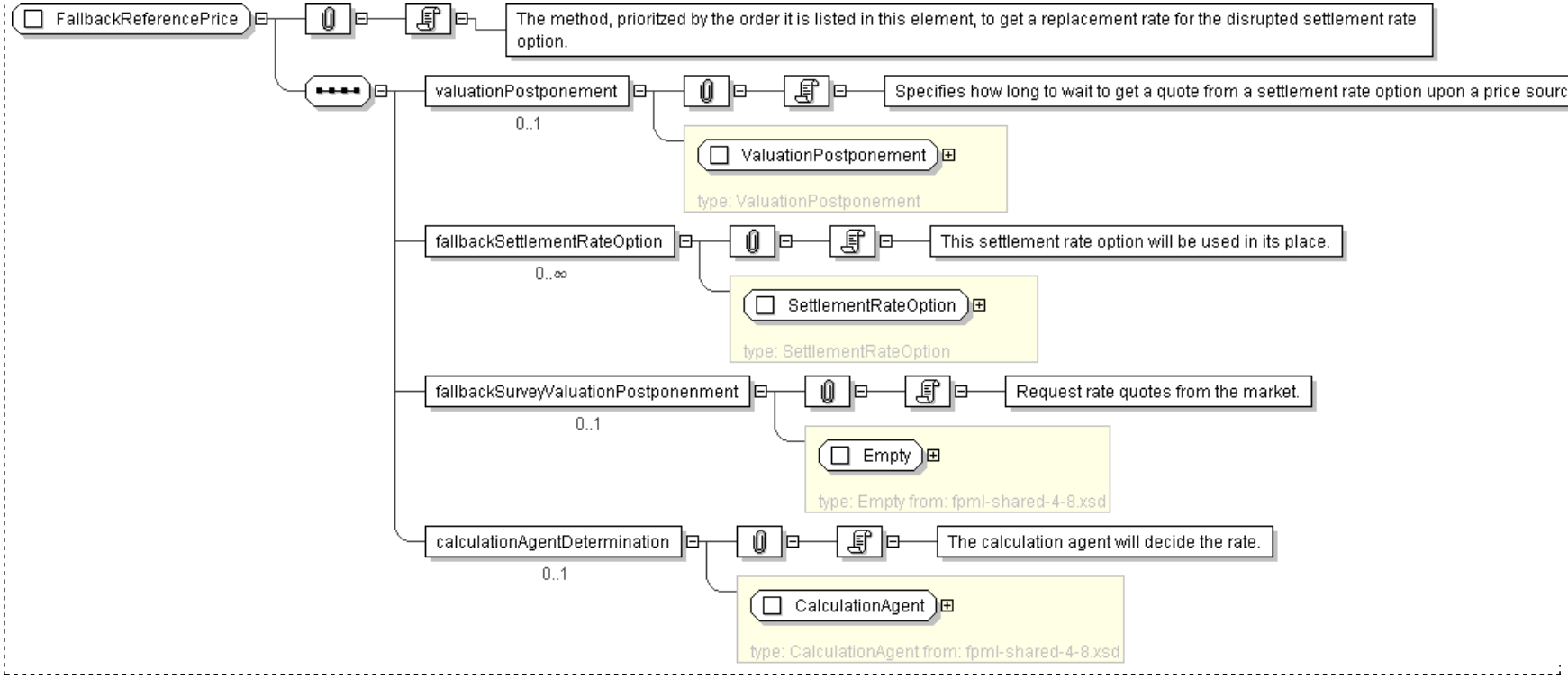
Super-types:	None
Sub-types:	None

Name	FallbackReferencePrice
Used by (from the same schema document)	Complex Type PriceSourceDisruption
Abstract	no
Documentation	The method, prioritzed by the order it is listed in this element, to get a replacement rate for the disrupted settlement rate option.

XML Instance Representation

```
<...>
   <valuationPostponement> ValuationPostponement </valuationPostponement> [0..1]
   'Specifies how long to wait to get a quote from a settlement rate option upon a price source disruption'

   <fallbackSettlementRateOption> SettlementRateOption </fallbackSettlementRateOption> [0..*]
   'This settlement rate option will be used in its place.'

   <fallbackSurveyValuationPostponenment> Empty </fallbackSurveyValuationPostponenment> [0..1]
   'Request rate quotes from the market.'

   <calculationAgentDetermination> CalculationAgent </calculationAgentDetermination> [0..1]
   'The calculation agent will decide the rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FallbackReferencePrice">
  <xsd:sequence>
    <xsd:element name="valuationPostponement" type=" ValuationPostponement " minOccurs="0"/>
    <xsd:element name="fallbackSettlementRateOption" type=" SettlementRateOption " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="fallbackSurveyValuationPostponenment" type=" Empty " minOccurs="0"/>
    <xsd:element name="calculationAgentDetermination" type=" CalculationAgent " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FinalCalculationPeriodDateAdjustment

[Table of contents]

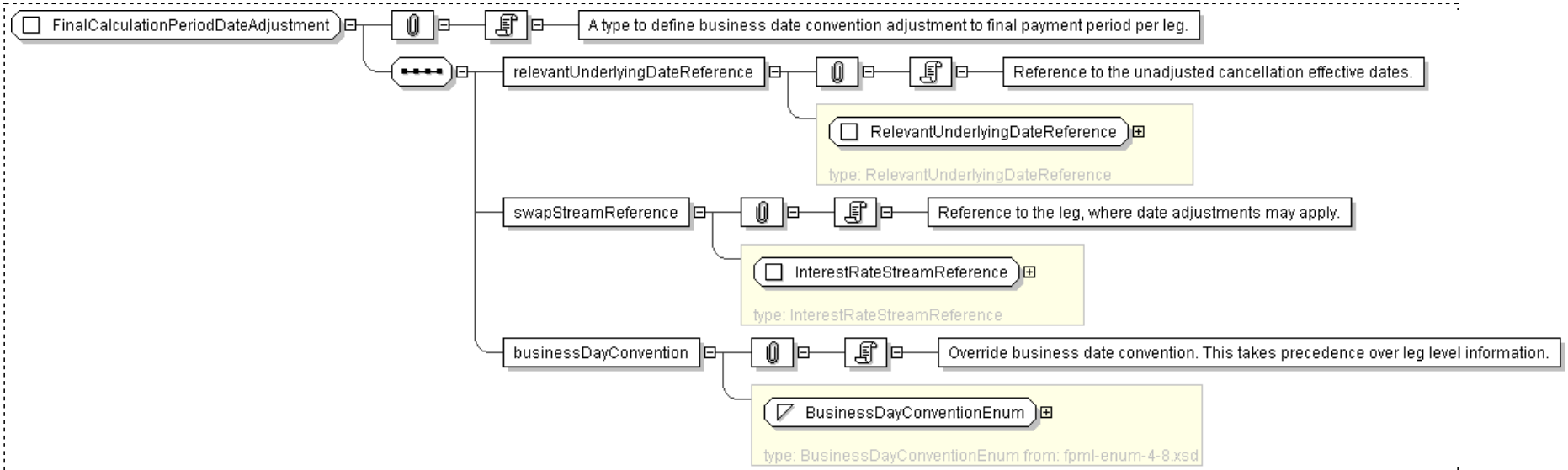
Super-types:	None
Sub-types:	None

Name	FinalCalculationPeriodDateAdjustment
Used by (from the same schema document)	Complex Type CancelableProvision
Abstract	no
Documentation	A type to define business date convention adjustment to final payment period per leg.

XML Instance Representation

```
<...>
  <relevantUnderlyingDateReference> RelevantUnderlyingDateReference </relevantUnderlyingDateReference> [1]
  'Reference to the unadjusted cancellation effective dates.'

  <swapStreamReference> InterestRateStreamReference </swapStreamReference> [1]
  'Reference to the leg, where date adjustments may apply.'

  <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
  'Override business date convention. This takes precedence over leg level information.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FinalCalculationPeriodDateAdjustment">
  <xsd:sequence>
    <xsd:element name="relevantUnderlyingDateReference" type=" RelevantUnderlyingDateReference "/>
    <xsd:element name="swapStreamReference" type=" InterestRateStreamReference "/>
```

```
    <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingRateDefinition

[Table of contents]

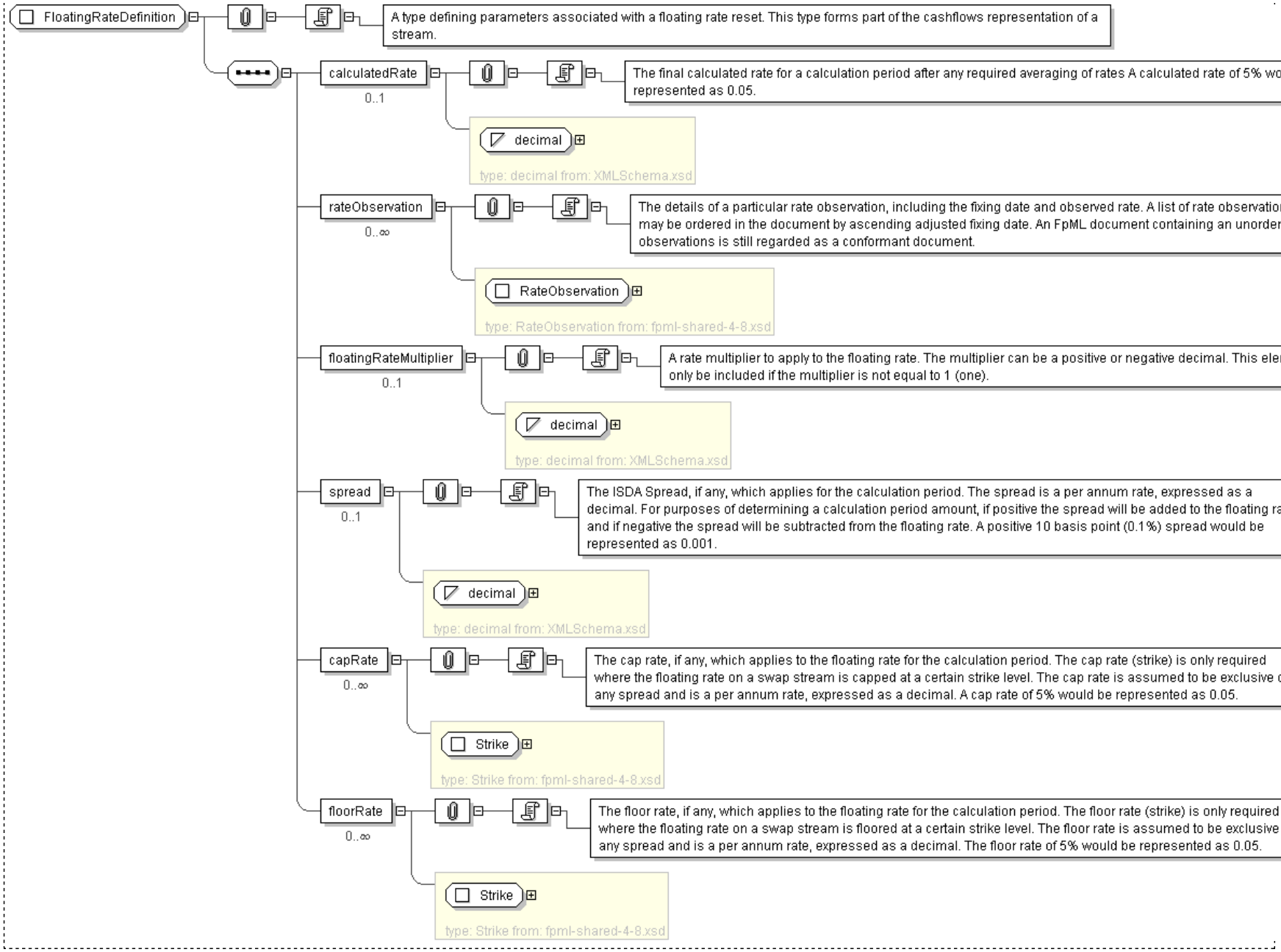
Super-types:	None
Sub-types:	None

Name	FloatingRateDefinition
Used by (from the same schema document)	Complex Type CalculationPeriod
Abstract	no
Documentation	A type defining parameters associated with a floating rate reset. This type forms part of the cashflows representation of a stream.

XML Instance Representation

```
<...>
  <calculatedRate> xsd:decimal </calculatedRate> [0..1]
  'The final calculated rate for a calculation period after any required averaging of rates A calculated rate of 5% would be represented as
  0.05.'

  <rateObservation> RateObservation </rateObservation> [0..*]
  'The details of a particular rate observation, including the fixing date and observed rate. A list of rate observation elements may be
  ordered in the document by ascending adjusted fixing date. An FpML document containing an unordered list of rate observations is still
  regarded as a conformant document.'

  <floatingRateMultiplier> xsd:decimal </floatingRateMultiplier> [0..1]
  'A rate multiplier to apply to the floating rate. The multiplier can be a positive or negative decimal. This element should only be included
  if the multiplier is not equal to 1 (one).'

  <spread> xsd:decimal </spread> [0..1]
  'The ISDA Spread, if any, which applies for the calculation period. The spread is a per annum rate, expressed as a decimal. For purposes of
  determining a calculation period amount, if positive the spread will be added to the floating rate and if negative the spread will be
  subtracted from the floating rate. A positive 10 basis point (0.1%) spread would be represented as 0.001.'

  <capRate> Strike </capRate> [0..*]
  'The cap rate, if any, which applies to the floating rate for the calculation period. The cap rate (strike) is only required where the
  floating rate on a swap stream is capped at a certain strike level. The cap rate is assumed to be exclusive of any spread and is a per annum
  rate, expressed as a decimal. A cap rate of 5% would be represented as 0.05.'

  <floorRate> Strike </floorRate> [0..*]
  'The floor rate, if any, which applies to the floating rate for the calculation period. The floor rate (strike) is only required where the
  floating rate on a swap stream is floored at a certain strike level. The floor rate is assumed to be exclusive of any spread and is a per
  annum rate, expressed as a decimal. The floor rate of 5% would be represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingRateDefinition">
  <xsd:sequence>
    <xsd:element name="calculatedRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="rateObservation" type=" RateObservation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="floatingRateMultiplier" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="spread" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="capRate" type=" Strike " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="floorRate" type=" Strike " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Fra

[Table of contents]

Super-types:	Product < Fra (by extension)
Sub-types:	None

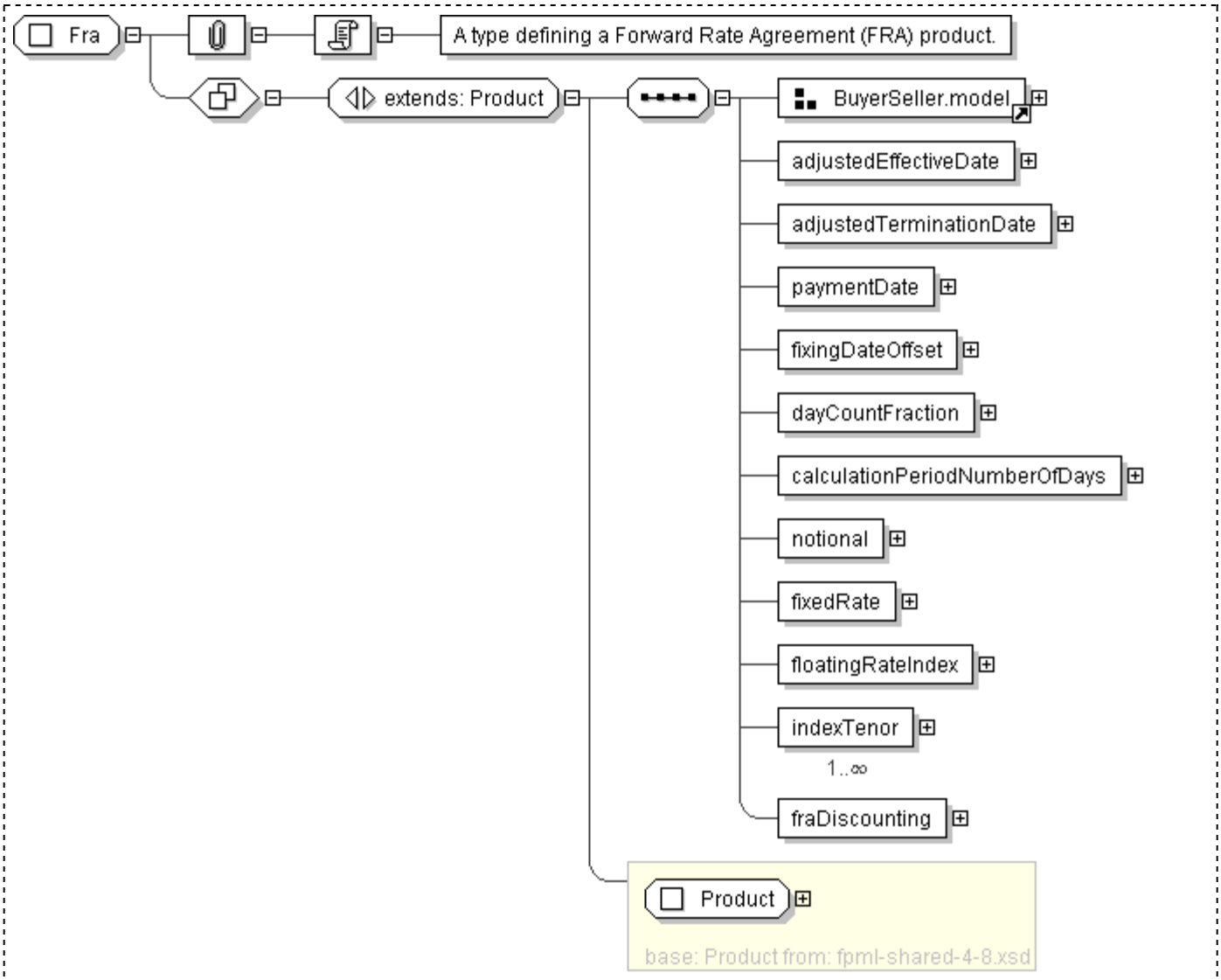
Name	Fra
Used by (from the same schema document)	Element fra
Abstract	no
Documentation	A type defining a Forward Rate Agreement (FRA) product.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this
    instrument and receives the rights defined by it. See 2000 ISDA definitions
    Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that
    grants the rights defined by this instrument and in return receives a payment
    for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is
    the floating rate payer.'

   <adjustedEffectiveDate> RequiredIdentifierDate </adjustedEffectiveDate> [1]
    'The start date of the calculation period. This date should already be adjusted
    for any applicable business day convention. This is also the date when the
    observed rate is applied, the reset date.'

   <adjustedTerminationDate> xsd:date </adjustedTerminationDate> [1]
    'The end date of the calculation period. This date should already be adjusted
    for any applicable business day convention.'

   <paymentDate> AdjustableDate </paymentDate> [1]
    'The payment date. This date is subject to adjustment in accordance with any
    applicable business day convention.'

   <fixingDateOffset> RelativeDateOffset </fixingDateOffset> [1]
    'Specifies the fixing date relative to the reset date in terms of a business
    days offset and an associated set of financial business centers. Normally these
    offset calculation rules will be those specified in the ISDA definition for the
    relevant floating rate index (ISDA\'s Floating Rate Option). However, non-
    standard offset calculation rules may apply for a trade if mutually agreed by
    the principal parties to the transaction. The href attribute on the
    dateRelativeTo element should reference the id attribute on the
    adjustedEffectiveDate element.'
```

```
<dayCountFraction> DayCountFraction </dayCountFraction> [1]
'The day count fraction.'

<calculationPeriodNumberOfDays> xsd:positiveInteger </calculationPeriodNumberOfDays>
[1]
'The number of days from the adjusted effective date to the adjusted
termination date calculated in accordance with the applicable day count
fraction.'

<notional> Money </notional> [1]
'The notional amount.'

<fixedRate> xsd:decimal </fixedRate> [1]
'The calculation period fixed rate. A per annum rate, expressed as a decimal. A
fixed rate of 5% would be represented as 0.05.'

<floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
<indexTenor> Period </indexTenor> [1..*]
'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

<fraDiscounting> FraDiscountingEnum </fraDiscounting> [1]
'Specifies whether discounting applies and, if so, what type.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Fra">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:group ref=" BuyerSeller.model "/>
            <xsd:element name="adjustedEffectiveDate" type=" RequiredIdentifierDate "/>
            <xsd:element name="adjustedTerminationDate" type=" xsd:date "/>
            <xsd:element name="paymentDate" type=" AdjustableDate "/>
            <xsd:element name="fixingDateOffset" type=" RelativeDateOffset "/>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "/>
            <xsd:element name="calculationPeriodNumberOfDays" type=" xsd:positiveInteger
            "/>
            <xsd:element name="notional" type=" Money "/>
            <xsd:element name="fixedRate" type=" xsd:decimal "/>
            <xsd:element name="floatingRateIndex" type=" FloatingRateIndex "/>
            <xsd:element name="indexTenor" type=" Period " maxOccurs="unbounded"/>
            <xsd:element name="fraDiscounting" type=" FraDiscountingEnum "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxFixingDate

[Table of contents]

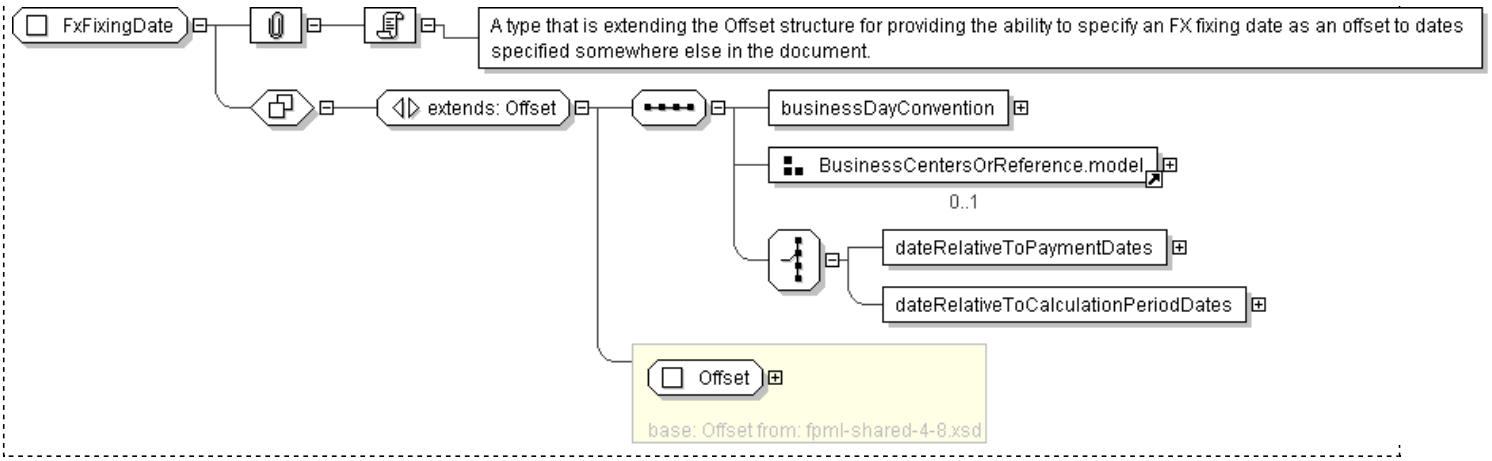
Super-types:	Offset < FxFixingDate (by extension)
Sub-types:	None

Name	FxFixingDate
Used by (from the same schema document)	Complex Type NonDeliverableSettlement
Abstract	no
Documentation	A type that is extending the Offset structure for providing the ability to specify an FX fixing date as an offset to dates specified somewhere else in the document.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset
   relative to another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0
   (zero) then period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
   'In the case of an offset specified as a number of days, this element defines whether consideration is
   given as to whether a day is a good business day or not. If a day type of business days is specified
   then non-business days are ignored when calculating the offset. The financial business centers to use
   for determination of business days are implied by the context in which this element is used. This
   element must only be included when the offset is specified as a number of days. If the offset is zero
   days then the dayType element should not be included.'

   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
   'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
         'A pointer style reference to a set of financial business centers defined elsewhere in the
         document. This set of business centers is used to determine whether a particular day is a business
         day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
   Start Choice [1]
      <dateRelativeToPaymentDates> DateRelativeToPaymentDates </dateRelativeToPaymentDates> [1]
      'The payment date references on which settlements in non-deliverable currency are due and will then
      have to be converted according to the terms specified through the other parts of the
      nonDeliverableSettlement structure.'

      <dateRelativeToCalculationPeriodDates> DateRelativeToCalculationPeriodDates
      </dateRelativeToCalculationPeriodDates> [1]
      'The calculation period references on which settlements in non-deliverable currency are due and will
      then have to be converted according to the terms specified through the other parts of the
      nonDeliverableSettlement structure. Implemented for Brazilian-CDI swaps where it will refer to the
      termination date of the appropriate leg.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxFixingDate">
  <xsd:complexContent>
    <xsd:extension base=" Offset ">
      <xsd:sequence>
        <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
        <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="dateRelativeToPaymentDates" type=" DateRelativeToPaymentDates "/>
          <xsd:element name="dateRelativeToCalculationPeriodDates" type="
          DateRelativeToCalculationPeriodDates "/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxLinkedNotionalAmount

[Table of contents]

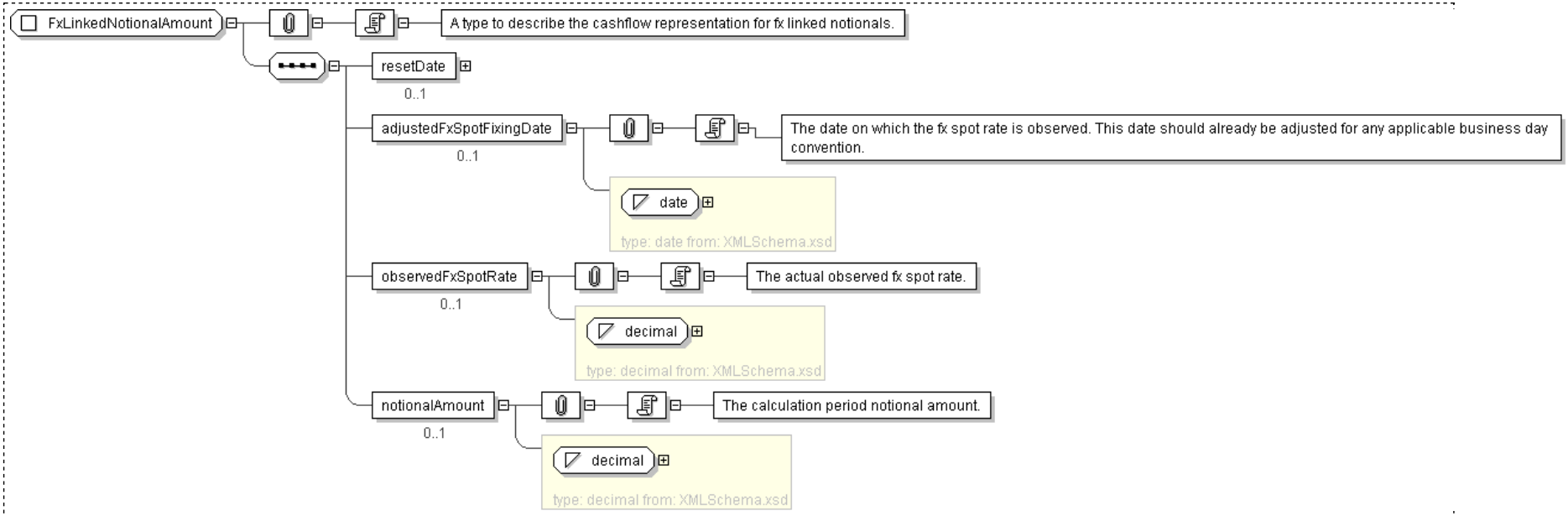
Super-types:	None
Sub-types:	None

Name	FxLinkedNotionalAmount
Used by (from the same schema document)	Complex Type CalculationPeriod
Abstract	no
Documentation	A type to describe the cashflow representation for fx linked notionals.

XML Instance Representation

```
<...>
   <resetDate> xsd:date </resetDate> [0..1]
   <adjustedFxSpotFixingDate> xsd:date </adjustedFxSpotFixingDate> [0..1]
   'The date on which the fx spot rate is observed. This date should already be adjusted for any applicable business day convention.'

   <observedFxSpotRate> xsd:decimal </observedFxSpotRate> [0..1]
   'The actual observed fx spot rate.'

   <notionalAmount> xsd:decimal </notionalAmount> [0..1]
   'The calculation period notional amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxLinkedNotionalAmount">
   <xsd:sequence>
      <xsd:element name="resetDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="adjustedFxSpotFixingDate" type=" xsd:date " minOccurs="0"/>
```

```
      <xsd:element name="observedFxSpotRate" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="notionalAmount" type=" xsd:decimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxLinkedNotionalSchedule

[Table of contents]

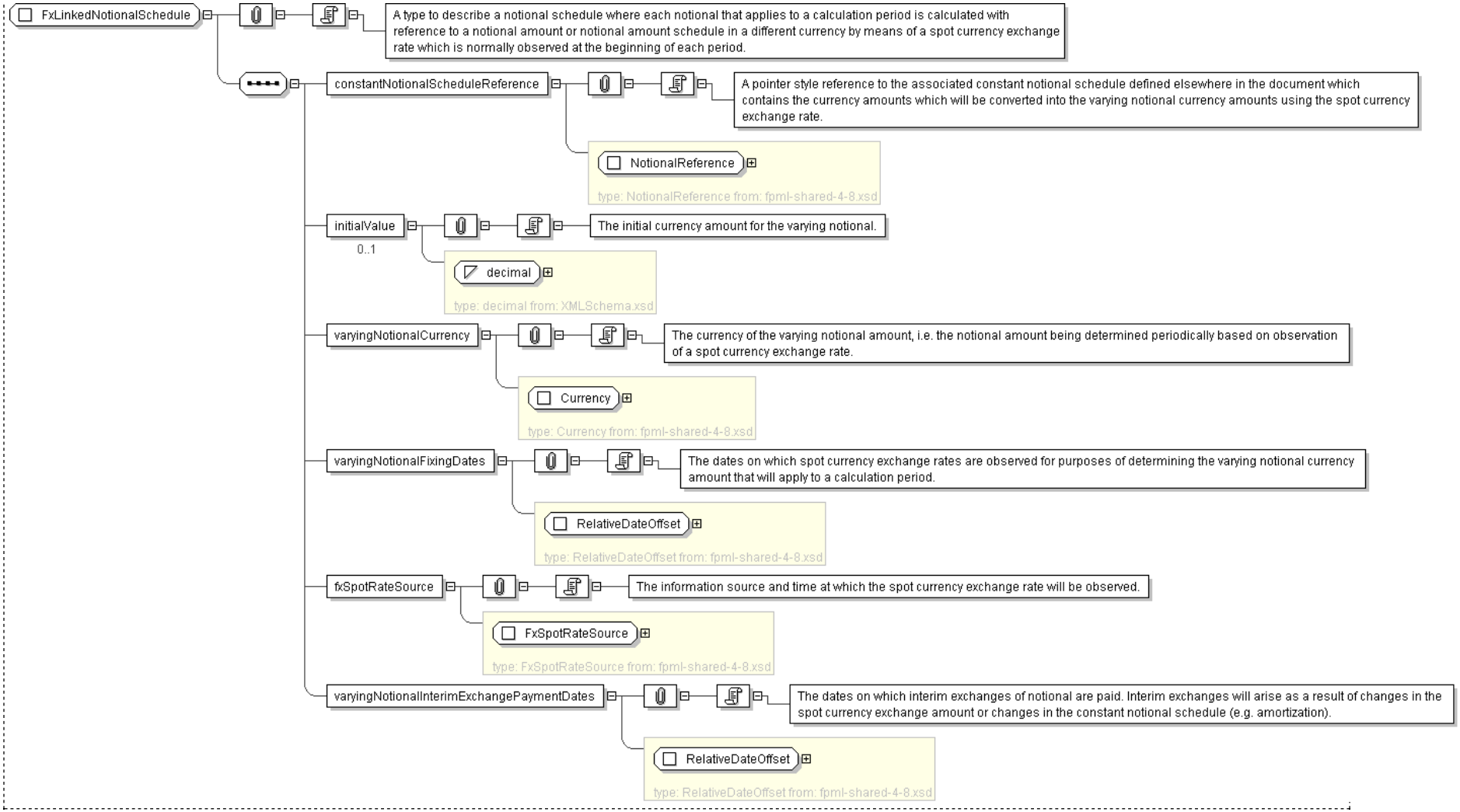
Super-types:	None
Sub-types:	None

Name	FxLinkedNotionalSchedule
Used by (from the same schema document)	Complex Type Calculation
Abstract	no
Documentation	A type to describe a notional schedule where each notional that applies to a calculation period is calculated with reference to a notional amount or notional amount schedule in a different currency by means of a spot currency exchange rate which is normally observed at the beginning of each period.

XML Instance Representation

```
<...>
   <constantNotionalScheduleReference> NotionalReference </constantNotionalScheduleReference> [1]
   'A pointer style reference to the associated constant notional schedule defined elsewhere in the document which contains the currency amounts which will be
   converted into the varying notional currency amounts using the spot currency exchange rate.'

   <initialValue> xsd:decimal </initialValue> [0..1]
   'The initial currency amount for the varying notional.'

   <varyingNotionalCurrency> Currency </varyingNotionalCurrency> [1]
   'The currency of the varying notional amount, i.e. the notional amount being determined periodically based on observation of a spot currency exchange rate.'

   <varyingNotionalFixingDates> RelativeDateOffset </varyingNotionalFixingDates> [1]
   'The dates on which spot currency exchange rates are observed for purposes of determining the varying notional currency amount that will apply to a calculation
   period.'

   <fxSpotRateSource> FxSpotRateSource </fxSpotRateSource> [1]
   'The information source and time at which the spot currency exchange rate will be observed.'

   <varyingNotionalInterimExchangePaymentDates> RelativeDateOffset </varyingNotionalInterimExchangePaymentDates> [1]
   'The dates on which interim exchanges of notional are paid. Interim exchanges will arise as a result of changes in the spot currency exchange amount or changes
   in the constant notional schedule (e.g. amortization).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxLinkedNotionalSchedule">
  <xsd:sequence>
    <xsd:element name="constantNotionalScheduleReference" type=" NotionalReference "/>
    <xsd:element name="initialValue" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="varyingNotionalCurrency" type=" Currency "/>
    <xsd:element name="varyingNotionalFixingDates" type=" RelativeDateOffset "/>
    <xsd:element name="fxSpotRateSource" type=" FxSpotRateSource "/>
    <xsd:element name="varyingNotionalInterimExchangePaymentDates" type=" RelativeDateOffset "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InflationRateCalculation

[Table of contents]

Super-types:	FloatingRateCalculation < InflationRateCalculation (by extension)
Sub-types:	None

Name	InflationRateCalculation
Used by (from the same schema document)	Element inflationRateCalculation
Abstract	no
Documentation	A type defining the components specifiying an Inflation Rate Calculation

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <floatingRateMultiplierSchedule> Schedule </floatingRateMultiplierSchedule> [0..1]
   'A rate multiplier or multiplier schedule to apply to the floating rate. A multiplier
   schedule is expressed as explicit multipliers and dates. In the case of a schedule, the step
   dates may be subject to adjustment in accordance with any adjustments specified in the
   calculationPeriodDatesAdjustments. The multiplier can be a positive or negative decimal. This
   element should only be included if the multiplier is not equal to 1 (one) for the term of
   the stream.'

   <spreadSchedule> SpreadSchedule </spreadSchedule> [0..*]
   'The ISDA Spread or a Spread schedule expressed as explicit spreads and dates. In the case
   of a schedule, the step dates may be subject to adjustment in accordance with any
   adjustments specified in calculationPeriodDatesAdjustments. The spread is a per annum rate,
   expressed as a decimal. For purposes of determining a calculation period amount, if positive
   the spread will be added to the floating rate and if negative the spread will be subtracted
   from the floating rate. A positive 10 basis point (0.1%) spread would be represented as
   0.001.'

   <rateTreatment> RateTreatmentEnum </rateTreatment> [0..1]
   'The specification of any rate conversion which needs to be applied to the observed rate
   before being used in any calculations. The two common conversions are for securities quoted
   on a bank discount basis which will need to be converted to either a Money Market Yield or
   Bond Equivalent Yield. See the Annex to the 2000 ISDA Definitions, Section 7.3. Certain
   General Definitions Relating to Floating Rate Options, paragraphs (g) and (h) for definitions
   of these terms.'

   <capRateSchedule> StrikeSchedule </capRateSchedule> [0..*]
   'The cap rate or cap rate schedule, if any, which applies to the floating rate. The cap rate
   (strike) is only required where the floating rate on a swap stream is capped at a certain
   level. A cap rate schedule is expressed as explicit cap rates and dates and the step dates
   may be subject to adjustment in accordance with any adjustments specified in
   calculationPeriodDatesAdjustments. The cap rate is assumed to be exclusive of any spread and
   is a per annum rate, expressed as a decimal. A cap rate of 5% would be represented as 0.05.'

   <floorRateSchedule> StrikeSchedule </floorRateSchedule> [0..*]
   'The floor rate or floor rate schedule, if any, which applies to the floating rate. The
   floor rate (strike) is only required where the floating rate on a swap stream is floored at
   a certain strike level. A floor rate schedule is expressed as explicit floor rates and dates
   and the step dates may be subject to adjustment in accordance with any adjustments specified
   in calculationPeriodDatesAdjustments. The floor rate is assumed to be exclusive of any spread
   and is a per annum rate, expressed as a decimal. A floor rate of 5% would be represented as
   0.05.'

   <initialRate> xsd:decimal </initialRate> [0..1]
   'The initial floating rate reset agreed between the principal parties involved in the trade.
   This is assumed to be the first required reset rate for the first regular calculation
   period. It should only be included when the rate is not equal to the rate published on the
   source implied by the floating rate index. An initial rate of 5% would be represented as
   0.05.'

   <finalRateRounding> Rounding </finalRateRounding> [0..1]
   'The rounding convention to apply to the final rate used in determination of a calculation
   period amount.'
```

<averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
'If averaging is applicable, this component specifies whether a weighted or unweighted average method of calculation is to be used. The component must only be included when averaging applies.'

<negativeInterestRateTreatment> NegativeInterestRateTreatmentEnum </negativeInterestRateTreatment> [0..1]
'The specification of any provisions for calculating payment obligations when a floating rate is negative (either due to a quoted negative floating rate or by operation of a spread that is subtracted from the floating rate).'

<inflationLag> Offset </inflationLag> [1]
'an offsetting period from the payment date which determines the reference period for which the inflation index is onserved.'

<indexSource> RateSourcePage </indexSource> [1]
'The reference source such as Reuters or Bloomberg.'

<mainPublication> MainPublication </mainPublication> [0..1]
'The current main publication source such as relevant web site or a government body.'

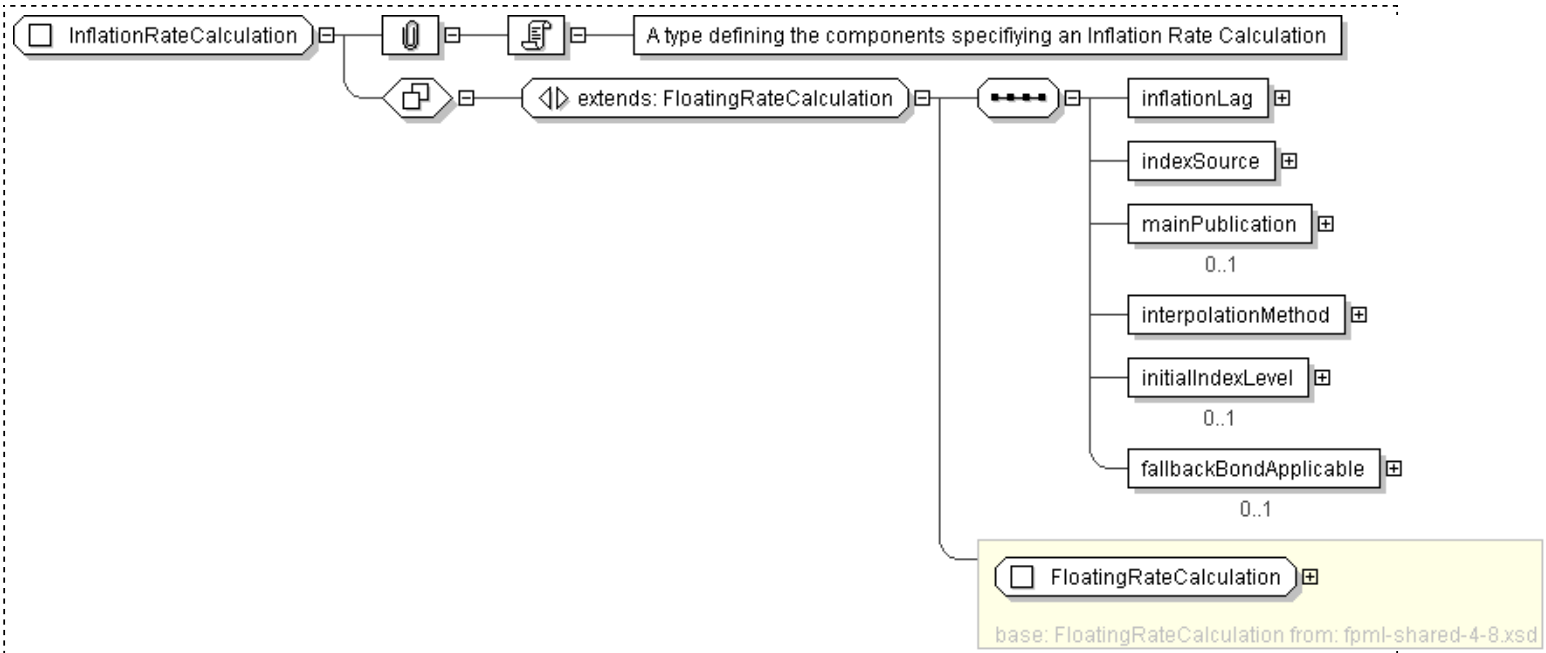
<interpolationMethod> InterpolationMethod </interpolationMethod> [1]
'The method used when calculating the Inflation Index Level from multiple points - the most common is Linear.'

<initialIndexLevel> xsd:decimal </initialIndexLevel> [0..1]
'initial known index level for the first calculation period.'

<fallbackBondApplicable> xsd:boolean </fallbackBondApplicable> [0..1]
'The applicability of a fallback bond as defined in the 2006 ISDA Inflation Derivatives Definitions, sections 1.3 and 1.8. Omission of this element imples a value of true.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="InflationRateCalculation">
  <xsd:complexContent>
    <xsd:extension base=" FloatingRateCalculation ">
      <xsd:sequence>
        <xsd:element name="inflationLag" type=" Offset "/>
        <xsd:element name="indexSource" type=" RateSourcePage "/>
        <xsd:element name="mainPublication" type=" MainPublication " minOccurs="0"/>
        <xsd:element name="interpolationMethod" type=" InterpolationMethod "/>
        <xsd:element name="initialIndexLevel" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="fallbackBondApplicable" type=" xsd:boolean " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestRateStream

[Table of contents]

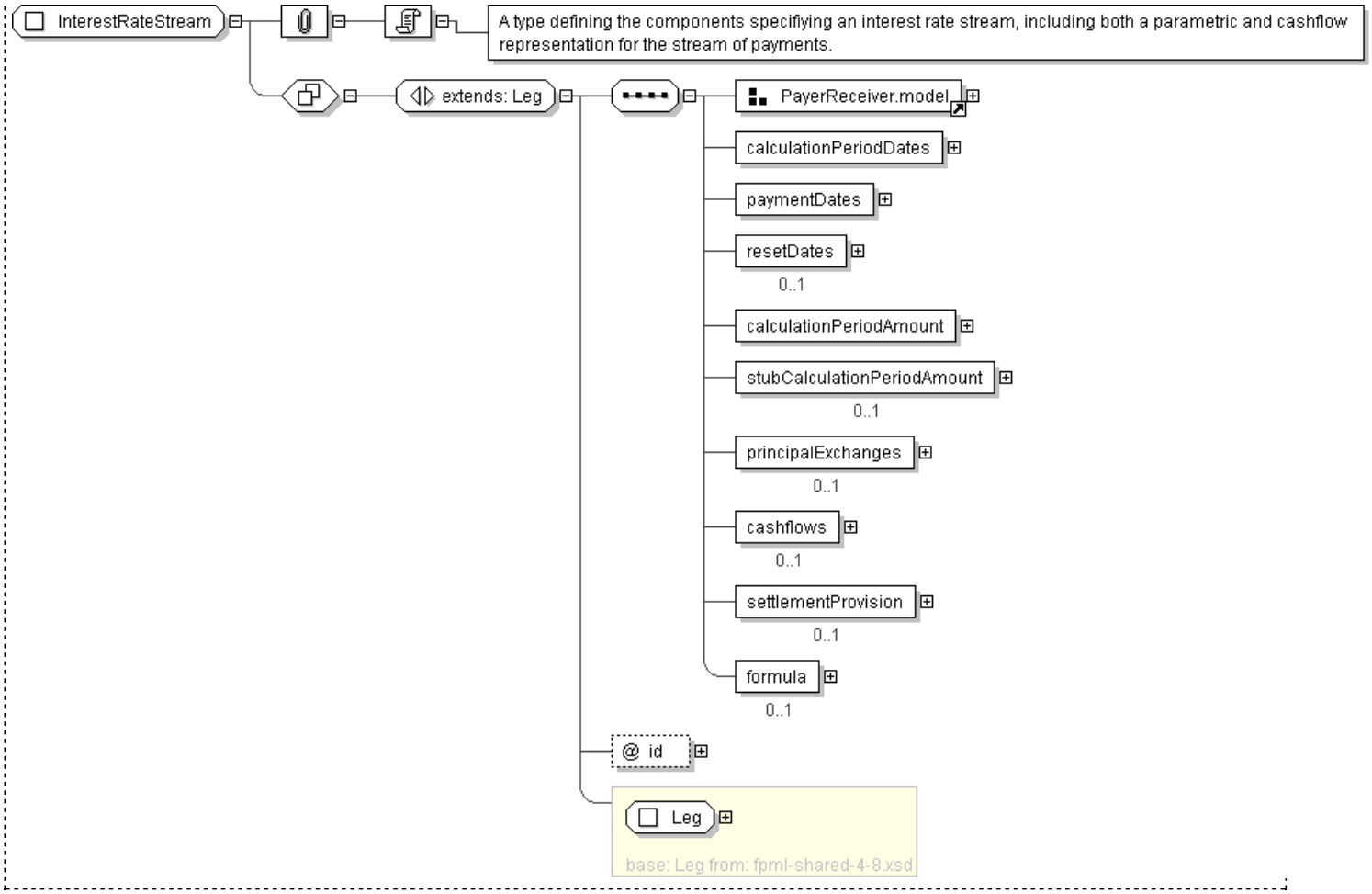
Super-types:	Leg < InterestRateStream (by extension)
Sub-types:	None

Name	InterestRateStream
Used by (from the same schema document)	Complex Type CapFloor , Complex Type Swap
Abstract	no
Documentation	A type defining the components specifiying an interest rate stream, including both a parametric and cashflow representation for the stream of payments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <calculationPeriodDates> CalculationPeriodDates </calculationPeriodDates> [1]
   'The calculation periods dates schedule.'

   <paymentDates> PaymentDates </paymentDates> [1]
   'The payment dates schedule.'

   <resetDates> ResetDates </resetDates> [0..1]
   'The reset dates schedule. The reset dates schedule only applies for a floating rate stream.'

   <calculationPeriodAmount> CalculationPeriodAmount </calculationPeriodAmount> [1]
   'The calculation period amount parameters.'

   <stubCalculationPeriodAmount> StubCalculationPeriodAmount </stubCalculationPeriodAmount> [0..1]
   'The stub calculation period amount parameters. This element must only be included if there is an
   initial or final stub calculation period. Even then, it must only be included if either the stub
   references a different floating rate tenor to the regular calculation periods, or if the stub is
   calculated as a linear interpolation of two different floating rate tenors, or if a specific stub rate
   or stub amount has been negotiated.'

   <principalExchanges> PrincipalExchanges </principalExchanges> [0..1]
   'The true/false flags indicating whether initial, intermediate or final exchanges of principal should
   occur.'

   <cashflows> Cashflows </cashflows> [0..1]
   'The cashflows representation of the swap stream.'

   <settlementProvision> SettlementProvision </settlementProvision> [0..1]
   'A provision that allows the specification of settlement terms, occuring when the settlement currency is
   different to the notional currency of the trade.'

   <formula> Formula </formula> [0..1]
   'An interest rate derivative formula.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestRateStream">
  <xsd:complexContent>
    <xsd:extension base=" Leg ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="calculationPeriodDates" type=" CalculationPeriodDates "/>
        <xsd:element name="paymentDates" type=" PaymentDates "/>
        <xsd:element name="resetDates" type=" ResetDates " minOccurs="0"/>
        <xsd:element name="calculationPeriodAmount" type=" CalculationPeriodAmount "/>
        <xsd:element name="stubCalculationPeriodAmount" type=" StubCalculationPeriodAmount " minOccurs="0"/>
        <xsd:element name="principalExchanges" type=" PrincipalExchanges " minOccurs="0"/>
        <xsd:element name="cashflows" type=" Cashflows " minOccurs="0"/>
        <xsd:element name="settlementProvision" type=" SettlementProvision " minOccurs="0"/>
        <xsd:element name="formula" type=" Formula " minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestRateStreamReference

[Table of contents]

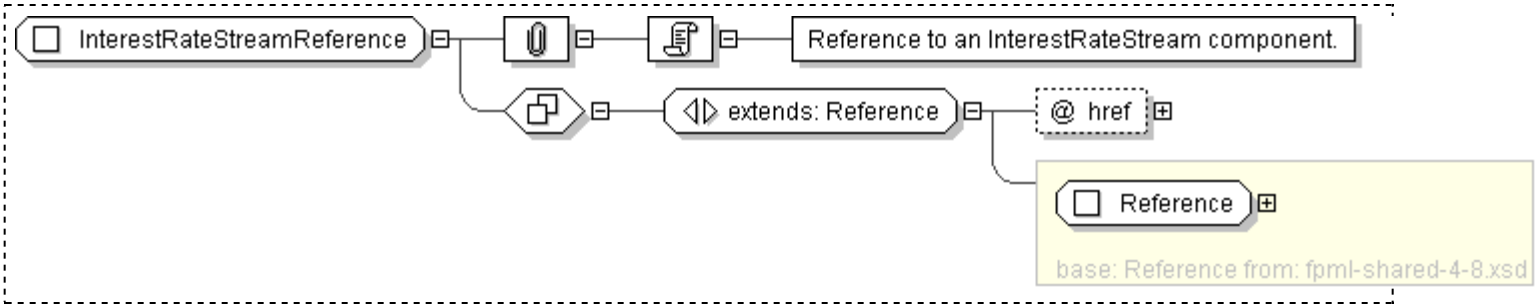
Super-types:	Reference < InterestRateStreamReference (by extension)
Sub-types:	None

Name	InterestRateStreamReference
Used by (from the same schema document)	Complex Type FinalCalculationPeriodDateAdjustment
Abstract	no
Documentation	Reference to an InterestRateStream component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestRateStreamReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="InterestRateStream"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MandatoryEarlyTermination

[Table of contents]

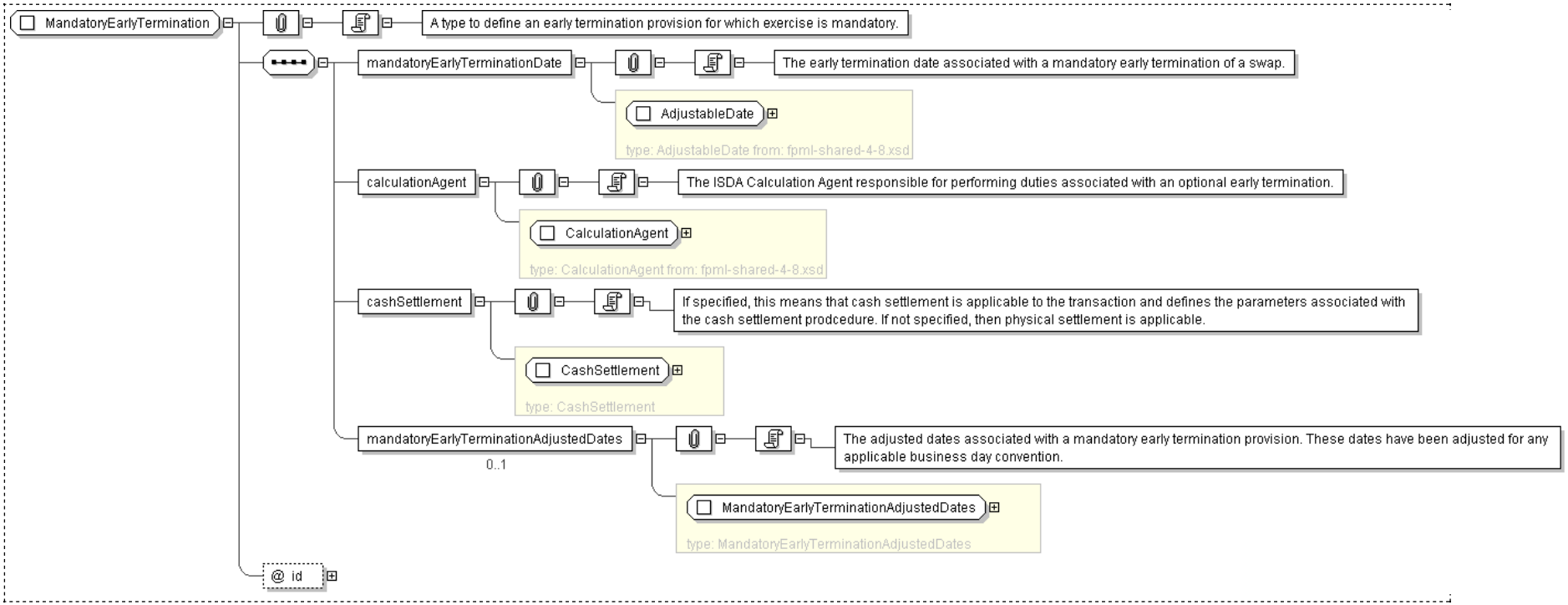
Super-types:	None
Sub-types:	None

Name	MandatoryEarlyTermination
Used by (from the same schema document)	Model Group MandatoryEarlyTermination.model , Model Group MandatoryEarlyTermination.model
Abstract	no
Documentation	A type to define an early termination provision for which exercise is mandatory.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <mandatoryEarlyTerminationDate> AdjustableDate </mandatoryEarlyTerminationDate> [1]
    'The early termination date associated with a mandatory early termination of a swap.'

   <calculationAgent> CalculationAgent </calculationAgent> [1]
    'The ISDA Calculation Agent responsible for performing duties associated with an optional early termination.'

   <cashSettlement> CashSettlement </cashSettlement> [1]
    'If specified, this means that cash settlement is applicable to the transaction and defines the parameters associated with the cash settlement prodcedure. If
    not specified, then physical settlement is applicable.'

   <mandatoryEarlyTerminationAdjustedDates> MandatoryEarlyTerminationAdjustedDates </mandatoryEarlyTerminationAdjustedDates> [0..1]
    'The adjusted dates associated with a mandatory early termination provision. These dates have been adjusted for any applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MandatoryEarlyTermination">
   <xsd:sequence>
      <xsd:element name="mandatoryEarlyTerminationDate" type=" AdjustableDate "/>
      <xsd:element name="calculationAgent" type=" CalculationAgent "/>
      <xsd:element name="cashSettlement" type=" CashSettlement "/>
      <xsd:element name="mandatoryEarlyTerminationAdjustedDates" type=" MandatoryEarlyTerminationAdjustedDates " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MandatoryEarlyTerminationAdjustedDates

[Table of contents]

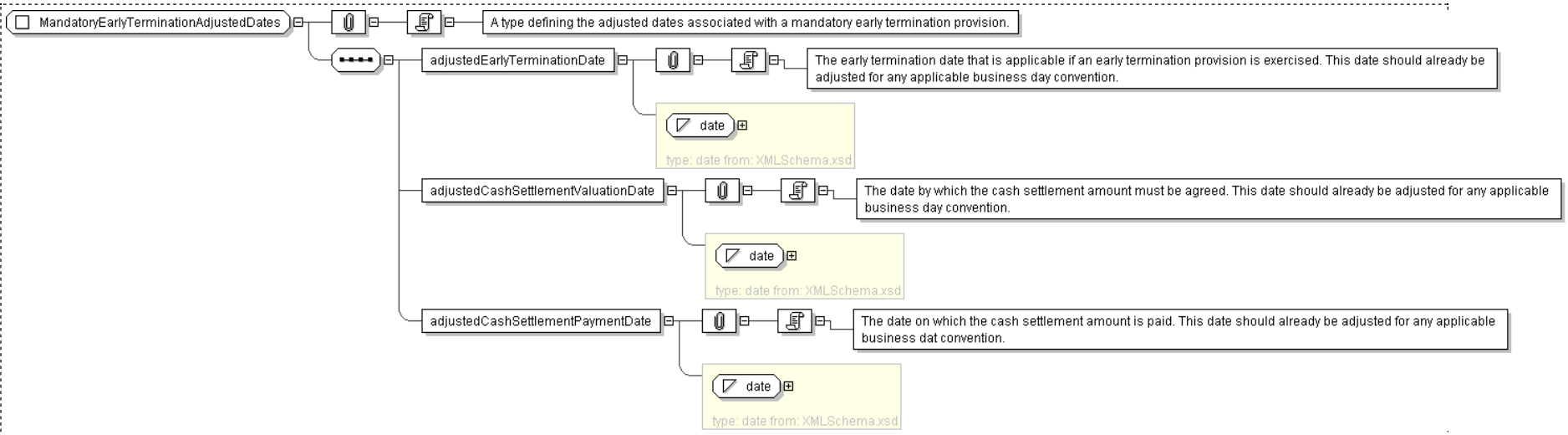
Super-types:	None
Sub-types:	None

Name	MandatoryEarlyTerminationAdjustedDates
Used by (from the same schema document)	Complex Type MandatoryEarlyTermination
Abstract	no
Documentation	A type defining the adjusted dates associated with a mandatory early termination provision.

XML Instance Representation

```
<...>
   <adjustedEarlyTerminationDate> xsd:date </adjustedEarlyTerminationDate> [1]
   'The early termination date that is applicable if an early termination provision is exercised. This date should already be adjusted for any applicable business day
   convention.'

   <adjustedCashSettlementValuationDate> xsd:date </adjustedCashSettlementValuationDate> [1]
   'The date by which the cash settlement amount must be agreed. This date should already be adjusted for any applicable business day convention.'

   <adjustedCashSettlementPaymentDate> xsd:date </adjustedCashSettlementPaymentDate> [1]
   'The date on which the cash settlement amount is paid. This date should already be adjusted for any applicable business dat convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MandatoryEarlyTerminationAdjustedDates">
   <xsd:sequence>
      <xsd:element name="adjustedEarlyTerminationDate" type=" xsd:date "/>
      <xsd:element name="adjustedCashSettlementValuationDate" type=" xsd:date "/>
      <xsd:element name="adjustedCashSettlementPaymentDate" type=" xsd:date "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonDeliverableSettlement

[Table of contents]

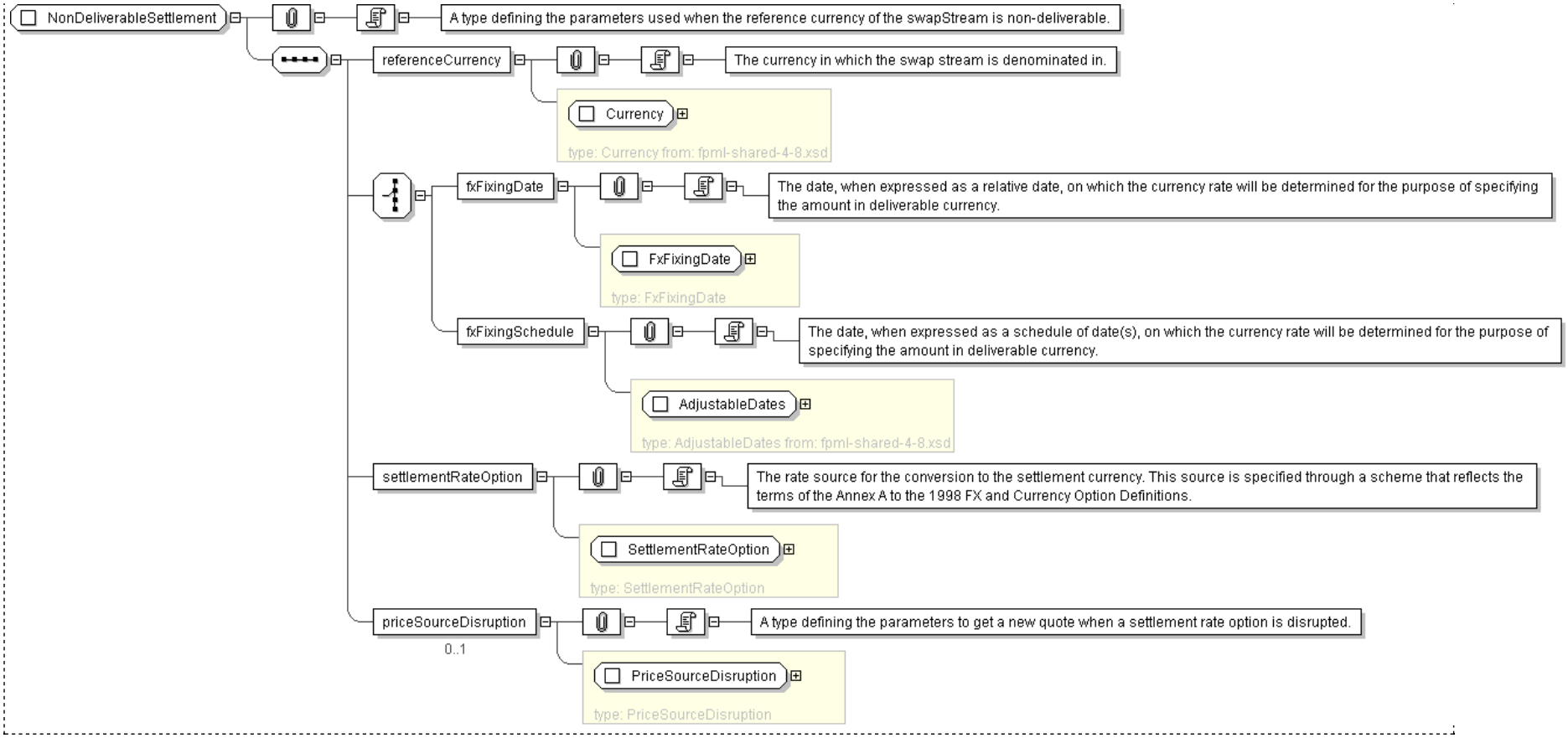
Super-types:	None
Sub-types:	None

Name	NonDeliverableSettlement
Used by (from the same schema document)	Complex Type SettlementProvision
Abstract	no
Documentation	A type defining the parameters used when the reference currency of the swapStream is non-deliverable.

XML Instance Representation

```
<...>
   <referenceCurrency> Currency </referenceCurrency> [1]
   'The currency in which the swap stream is denominated in.'

   Start Choice [1]
      <fxFixingDate> FxFixingDate </fxFixingDate> [1]
      'The date, when expressed as a relative date, on which the currency rate will be determined for the purpose of specifying the amount in
      deliverable currency.'

      <fxFixingSchedule> AdjustableDates </fxFixingSchedule> [1]
      'The date, when expressed as a schedule of date(s), on which the currency rate will be determined for the purpose of specifying the amount in
      deliverable currency.'

   End Choice
   <settlementRateOption> SettlementRateOption </settlementRateOption> [1]
   'The rate source for the conversion to the settlement currency. This source is specified through a scheme that reflects the terms of the Annex A to
   the 1998 FX and Currency Option Definitions.'

   <priceSourceDisruption> PriceSourceDisruption </priceSourceDisruption> [0..1]
   'A type defining the parameters to get a new quote when a settlement rate option is disrupted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonDeliverableSettlement">
  <xsd:sequence>
    <xsd:element name="referenceCurrency" type=" Currency "/>
    <xsd:choice>
      <xsd:element name="fxFixingDate" type=" FxFixingDate "/>
      <xsd:element name="fxFixingSchedule" type=" AdjustableDates "/>
    </xsd:choice>
    <xsd:element name="settlementRateOption" type=" SettlementRateOption "/>
    <xsd:element name="priceSourceDisruption" type=" PriceSourceDisruption " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Notional

[Table of contents]

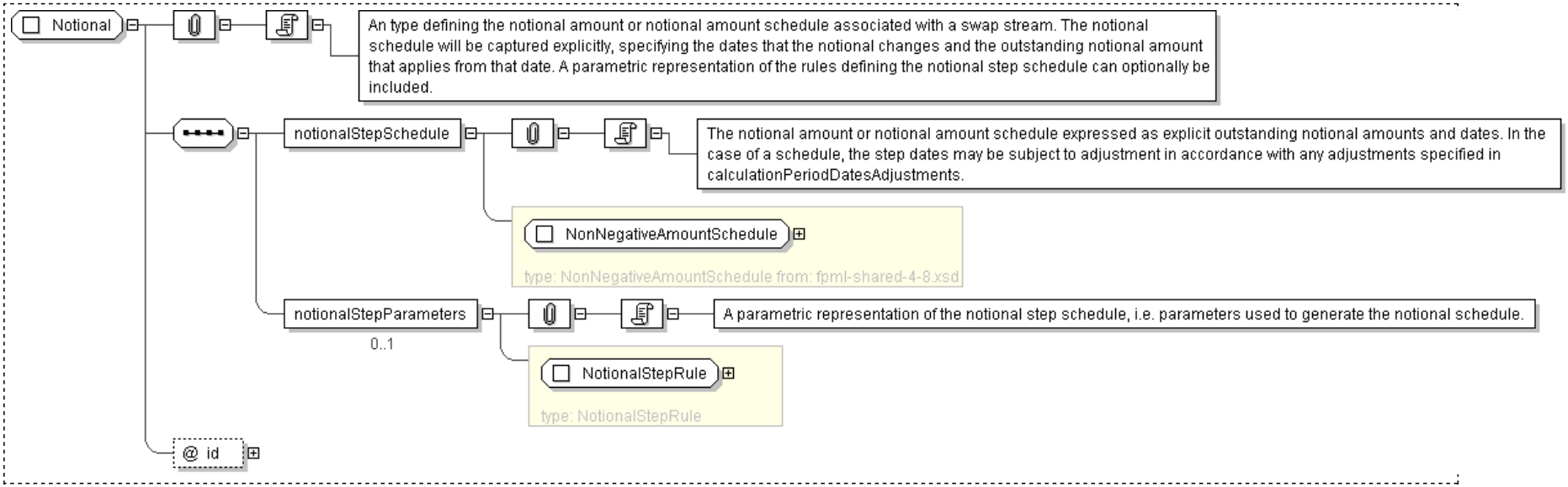
Super-types:	None
Sub-types:	None

Name	Notional
Used by (from the same schema document)	Complex Type Calculation
Abstract	no
Documentation	An type defining the notional amount or notional amount schedule associated with a swap stream. The notional schedule will be captured explicitly, specifying the dates that the notional changes and the outstanding notional amount that applies from that date. A parametric representation of the rules defining the notional step schedule can optionally be included.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <notionalStepSchedule> NonNegativeAmountSchedule </notionalStepSchedule> [1]
   'The notional amount or notional amount schedule expressed as explicit outstanding notional amounts and dates. In the case of a
   schedule, the step dates may be subject to adjustment in accordance with any adjustments specified in
   calculationPeriodDatesAdjustments.'

   <notionalStepParameters> NotionalStepRule </notionalStepParameters> [0..1]
   'A parametric representation of the notional step schedule, i.e. parameters used to generate the notional schedule.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Notional">
   <xsd:sequence>
```

```
    <xsd:element name="notionalStepSchedule" type=" NonNegativeAmountSchedule "/>
    <xsd:element name="notionalStepParameters" type=" NotionalStepRule " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotionalStepRule

[Table of contents]

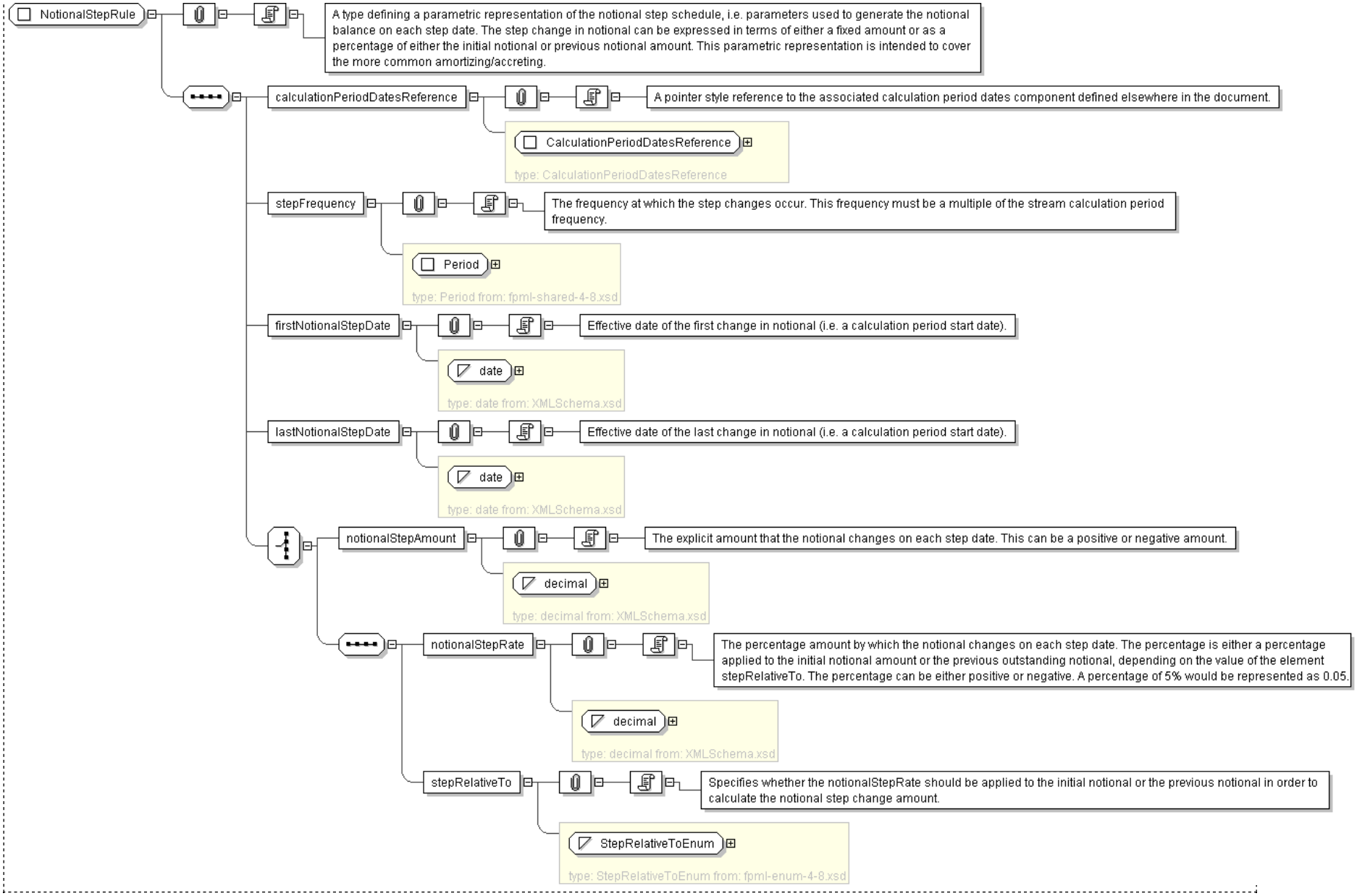
Super-types:	None
Sub-types:	None

Name	NotionalStepRule
Used by (from the same schema document)	Complex Type Notional
Abstract	no
Documentation	A type defining a parametric representation of the notional step schedule, i.e. parameters used to generate the notional balance on each step date. The step change in notional can be expressed in terms of either a fixed amount or as a percentage of either the initial notional or previous notional amount. This parametric representation is intended to cover the more common amortizing/accreting.

XML Instance Representation

```
<...>
   <calculationPeriodDatesReference> CalculationPeriodDatesReference </calculationPeriodDatesReference> [1]
   'A pointer style reference to the associated calculation period dates component defined elsewhere in the document.'

   <stepFrequency> Period </stepFrequency> [1]
   'The frequency at which the step changes occur. This frequency must be a multiple of the stream calculation period frequency.'

   <firstNotionalStepDate> xsd:date </firstNotionalStepDate> [1]
   'Effective date of the first change in notional (i.e. a calculation period start date).'

   <lastNotionalStepDate> xsd:date </lastNotionalStepDate> [1]
   'Effective date of the last change in notional (i.e. a calculation period start date).'

   Start Choice [1]
      <notionalStepAmount> xsd:decimal </notionalStepAmount> [1]
      'The explicit amount that the notional changes on each step date. This can be a positive or negative amount.'

      <notionalStepRate> xsd:decimal </notionalStepRate> [1]
      'The percentage amount by which the notional changes on each step date. The percentage is either a percentage applied to the initial notional amount
      or the previous outstanding notional, depending on the value of the element stepRelativeTo. The percentage can be either positive or negative. A
      percentage of 5% would be represented as 0.05.'

      <stepRelativeTo> StepRelativeToEnum </stepRelativeTo> [1]
      'Specifies whether the notionalStepRate should be applied to the initial notional or the previous notional in order to calculate the notional step
      change amount.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotionalStepRule">
  <xsd:sequence>
    <xsd:element name="calculationPeriodDatesReference" type=" CalculationPeriodDatesReference "/>
    <xsd:element name="stepFrequency" type=" Period "/>
    <xsd:element name="firstNotionalStepDate" type=" xsd:date "/>
    <xsd:element name="lastNotionalStepDate" type=" xsd:date "/>
    <xsd:choice>
      <xsd:element name="notionalStepAmount" type=" xsd:decimal "/>
      <xsd:sequence>
        <xsd:element name="notionalStepRate" type=" xsd:decimal "/>
        <xsd:element name="stepRelativeTo" type=" StepRelativeToEnum "/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionalEarlyTermination

[Table of contents]

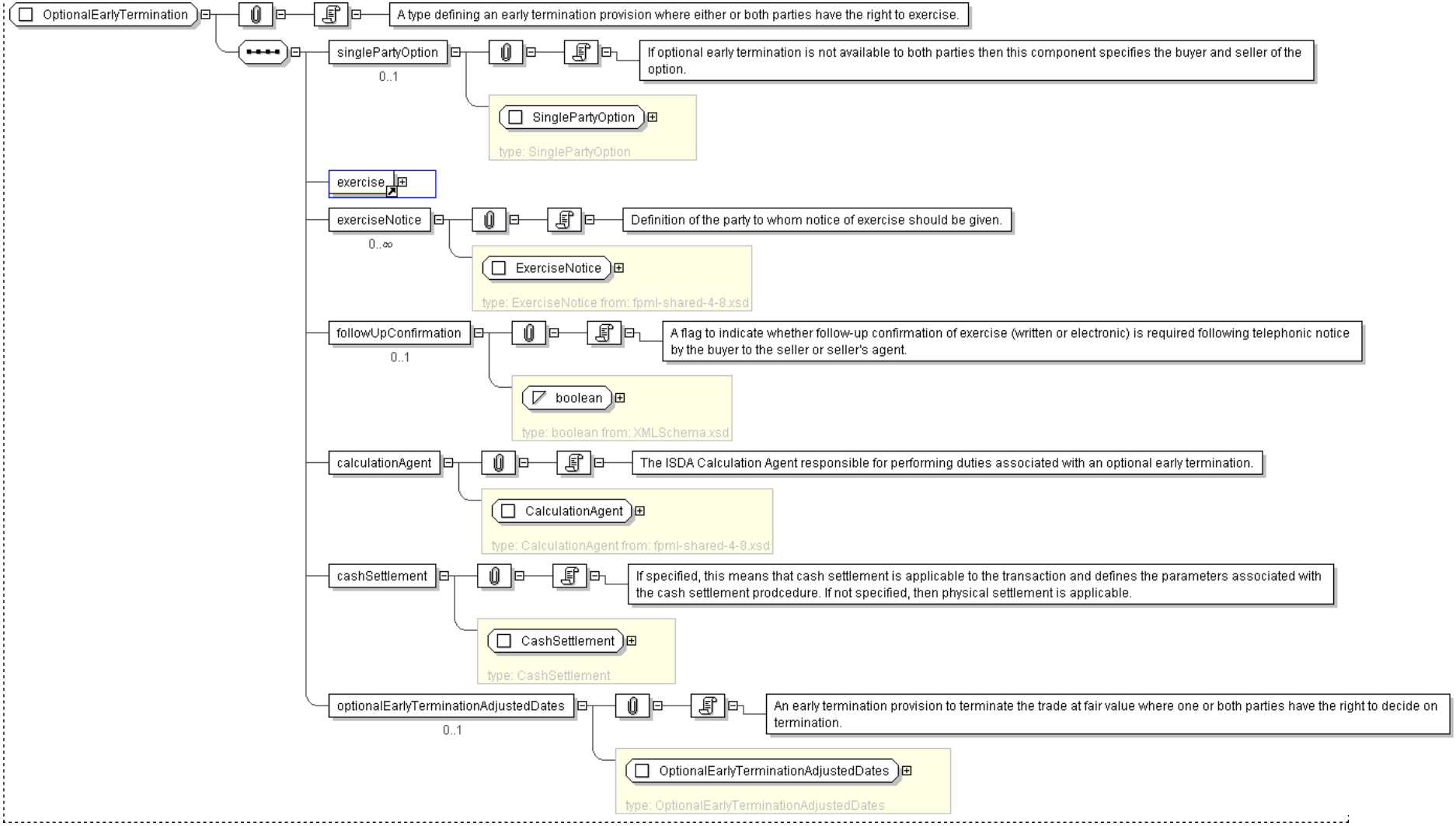
Super-types:	None
Sub-types:	None

Name	OptionalEarlyTermination
Used by (from the same schema document)	Model Group OptionalEarlyTermination.model , Model Group OptionalEarlyTermination.model
Abstract	no
Documentation	A type defining an early termination provision where either or both parties have the right to exercise.

XML Instance Representation

```
<...>
   <singlePartyOption> SinglePartyOption </singlePartyOption> [0..1]
   'If optional early termination is not available to both parties then this component specifies the buyer and seller of the option.'

   <exercise> ... </exercise> [1]
   <exerciseNotice> ExerciseNotice </exerciseNotice> [0..*]
   'Definition of the party to whom notice of exercise should be given.'

   <followUpConfirmation> xsd:boolean </followUpConfirmation> [0..1]
   'A flag to indicate whether follow-up confirmation of exercise (written or electronic) is required following telephonic notice by the buyer to the seller
   or seller\'s agent.'

   <calculationAgent> CalculationAgent </calculationAgent> [1]
   'The ISDA Calculation Agent responsible for performing duties associated with an optional early termination.'

   <cashSettlement> CashSettlement </cashSettlement> [1]
   'If specified, this means that cash settlement is applicable to the transaction and defines the parameters associated with the cash settlement prodcedure.
   If not specified, then physical settlement is applicable.'

   <optionalEarlyTerminationAdjustedDates> OptionalEarlyTerminationAdjustedDates </optionalEarlyTerminationAdjustedDates> [0..1]
   'An early termination provision to terminate the trade at fair value where one or both parties have the right to decide on termination.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionalEarlyTermination">
   <xsd:sequence>
      <xsd:element name="singlePartyOption" type=" SinglePartyOption " minOccurs="0"/>
      <xsd:element ref=" exercise "/>
      <xsd:element name="exerciseNotice" type=" ExerciseNotice " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="followUpConfirmation" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="calculationAgent" type=" CalculationAgent "/>
      <xsd:element name="cashSettlement" type=" CashSettlement "/>
      <xsd:element name="optionalEarlyTerminationAdjustedDates" type=" OptionalEarlyTerminationAdjustedDates " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionalEarlyTerminationAdjustedDates

[Table of contents]

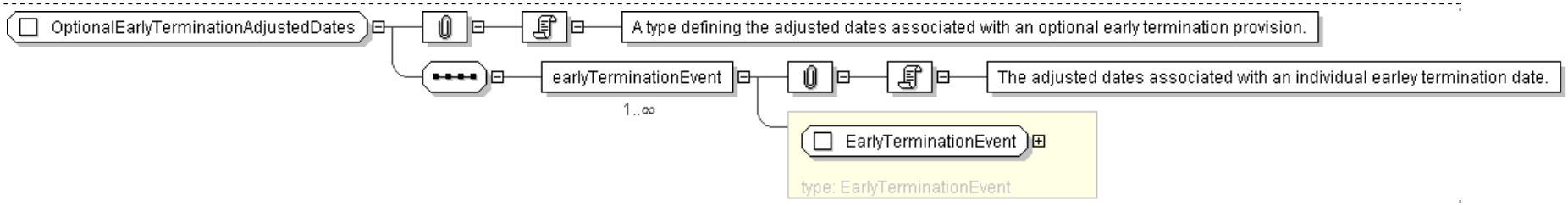
Super-types:	None
Sub-types:	None

Name	OptionalEarlyTerminationAdjustedDates
Used by (from the same schema document)	Complex Type OptionalEarlyTermination
Abstract	no
Documentation	A type defining the adjusted dates associated with an optional early termination provision.

XML Instance Representation

```
<...>
   <earlyTerminationEvent> EarlyTerminationEvent </earlyTerminationEvent> [1..*]
   'The adjusted dates associated with an individual earley termination date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionalEarlyTerminationAdjustedDates">
  <xsd:sequence>
    <xsd:element name="earlyTerminationEvent" type=" EarlyTerminationEvent " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentCalculationPeriod

[Table of contents]

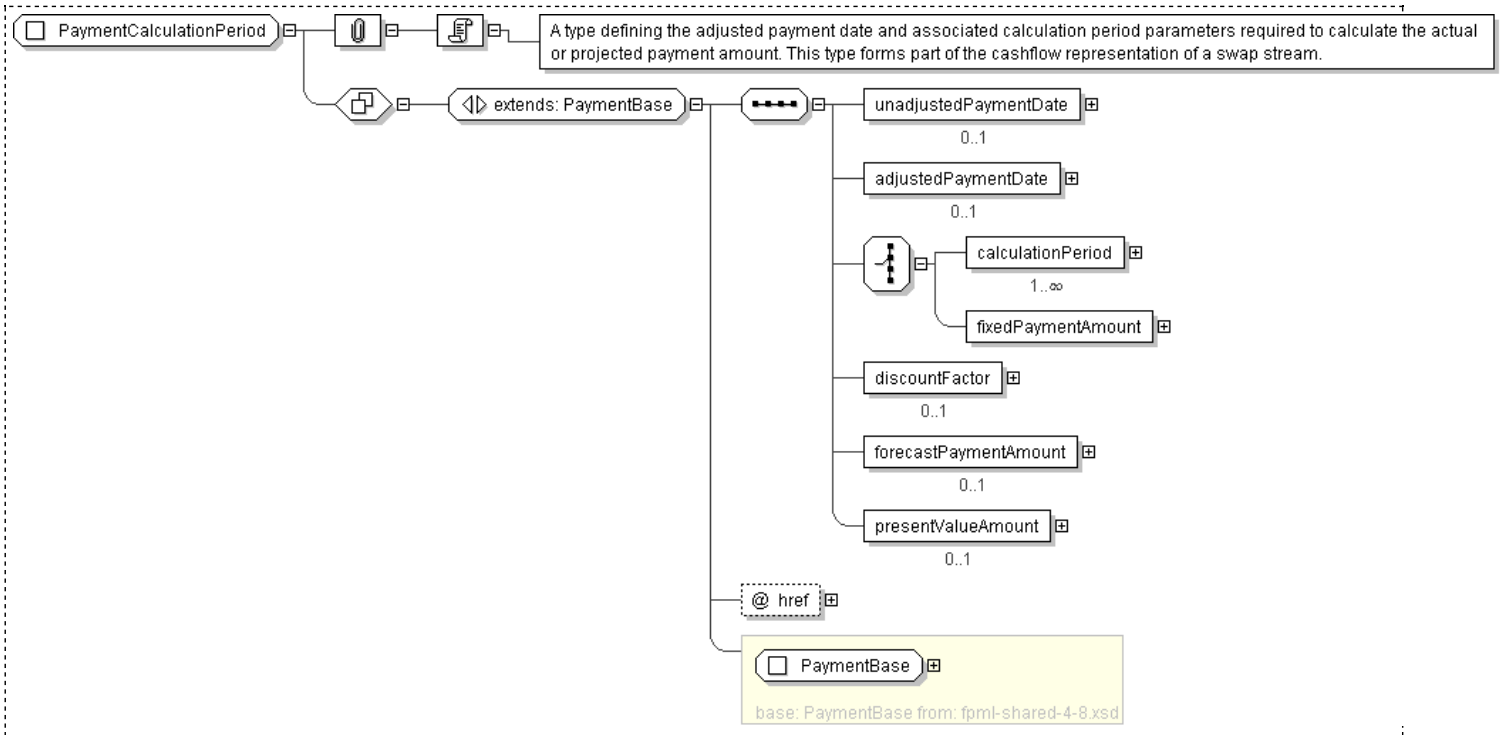
Super-types:	PaymentBase < PaymentCalculationPeriod (by extension)
Sub-types:	None

Name	PaymentCalculationPeriod
Used by (from the same schema document)	Complex Type Cashflows
Abstract	no
Documentation	A type defining the adjusted payment date and associated calculation period parameters required to calculate the actual or projected payment amount. This type forms part of the cashflow representation of a swap stream.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 href=" xsd:IDREF [0..1]
'Attribute that can be used to reference the yield curve used to estimate the discount factor.'

 ">
   <unadjustedPaymentDate> xsd:date </unadjustedPaymentDate> [0..1]
   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [0..1]
   'The adjusted payment date. This date should already be adjusted for any applicable business day convention. This
   component is not intended for use in trade confirmation but may be specified to allow the fee structure to also
   serve as a cashflow type component (all dates the Cashflows type are adjusted payment dates).'

   Start Choice [1]
      <calculationPeriod> CalculationPeriod </calculationPeriod> [1..*]
      'The parameters used in the calculation of a fixed or floating rate calculation period amount. A list of
      calculation period elements may be ordered in the document by ascending start date. An FpML document which
      contains an unordered list of calcularion periods is still regarded as a conformant document.'

      <fixedPaymentAmount> xsd:decimal </fixedPaymentAmount> [1]
      'A known fixed payment amount.'

   End Choice
   <discountFactor> xsd:decimal </discountFactor> [0..1]
   'A decimal value representing the discount factor used to calculate the present value of cash flow.'

   <forecastPaymentAmount> Money </forecastPaymentAmount> [0..1]
   'A monetary amount representing the forecast of the future value of the payment.'

   <presentValueAmount> Money </presentValueAmount> [0..1]
   'A monetary amount representing the present value of the forecast payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentCalculationPeriod">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="unadjustedPaymentDate" type=" xsd:date " minOccurs="0"/>
        <xsd:element name="adjustedPaymentDate" type=" xsd:date " minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="calculationPeriod" type=" CalculationPeriod " maxOccurs="unbounded"/>
          <xsd:element name="fixedPaymentAmount" type=" xsd:decimal "/>
        </xsd:choice>
        <xsd:element name="discountFactor" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="forecastPaymentAmount" type=" Money " minOccurs="0"/>
        <xsd:element name="presentValueAmount" type=" Money " minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="href" type=" xsd:IDREF " reference="PricingStructure"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentDates

[Table of contents]

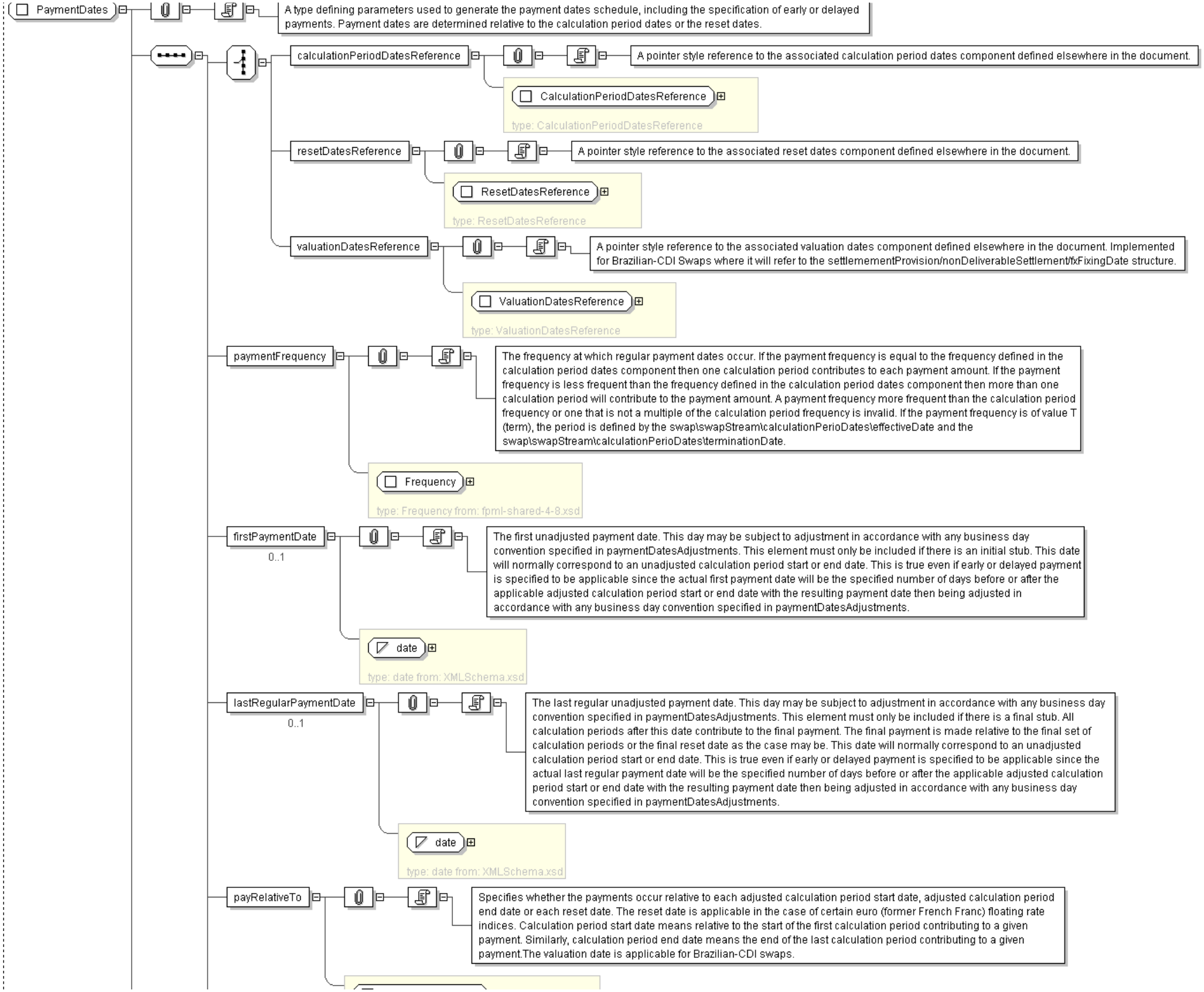
Super-types:	None
Sub-types:	None

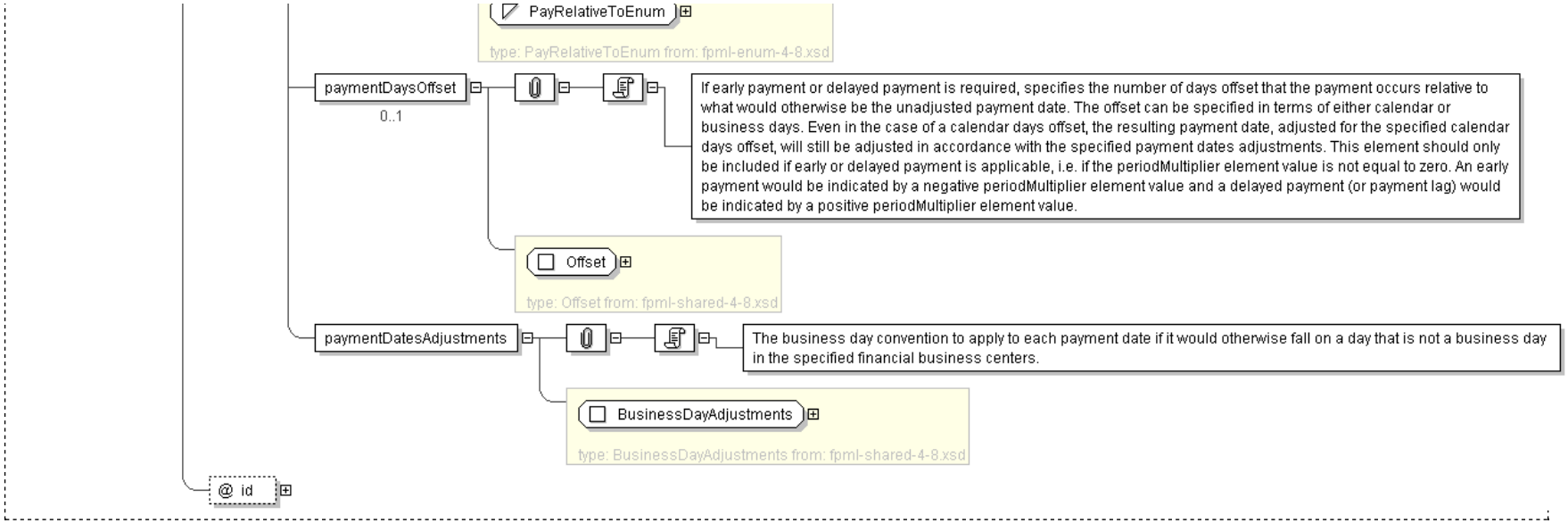
Name	PaymentDates
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining parameters used to generate the payment dates schedule, including the specification of early or delayed payments. Payment dates are determined relative to the calculation period dates or the reset dates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <calculationPeriodDatesReference> CalculationPeriodDatesReference </calculationPeriodDatesReference> [1]
       'A pointer style reference to the associated calculation period dates component defined elsewhere in the document.'

      <resetDatesReference> ResetDatesReference </resetDatesReference> [1]
       'A pointer style reference to the associated reset dates component defined elsewhere in the document.'

      <valuationDatesReference> ValuationDatesReference </valuationDatesReference> [1]
       'A pointer style reference to the associated valuation dates component defined elsewhere in the document. Implemented for Brazilian-CDI Swaps where
       it will refer to the settlemementProvision/nonDeliverableSettlement/fxFixingDate structure.'

   End Choice
   <paymentFrequency> Frequency </paymentFrequency> [1]
    'The frequency at which regular payment dates occur. If the payment frequency is equal to the frequency defined in the calculation period dates
    component then one calculation period contributes to each payment amount. If the payment frequency is less frequent than the frequency defined in the
    calculation period dates component then more than one calculation period will contribute to the payment amount. A payment frequency more frequent than
    the calculation period frequency or one that is not a multiple of the calculation period frequency is invalid. If the payment frequency is of value T
    (term), the period is defined by the swap\swapStream\calculationPerioDates\effectiveDate and the swap\swapStream\calculationPerioDates\terminationDate.'

   <firstPaymentDate> xsd:date </firstPaymentDate> [0..1]
    'The first unadjusted payment date. This day may be subject to adjustment in accordance with any business day convention specified in
    paymentDatesAdjustments. This element must only be included if there is an initial stub. This date will normally correspond to an unadjusted
    calculation period start or end date. This is true even if early or delayed payment is specified to be applicable since the actual first payment date
    will be the specified number of days before or after the applicable adjusted calculation period start or end date with the resulting payment date then
    being adjusted in accordance with any business day convention specified in paymentDatesAdjustments.'

   <lastRegularPaymentDate> xsd:date </lastRegularPaymentDate> [0..1]
    'The last regular unadjusted payment date. This day may be subject to adjustment in accordance with any business day convention specified in
    paymentDatesAdjustments. This element must only be included if there is a final stub. All calculation periods after this date contribute to the final
    payment. The final payment is made relative to the final set of calculation periods or the final reset date as the case may be. This date will
    normally correspond to an unadjusted calculation period start or end date. This is true even if early or delayed payment is specified to be applicable
    since the actual last regular payment date will be the specified number of days before or after the applicable adjusted calculation period start or
    end date with the resulting payment date then being adjusted in accordance with any business day convention specified in paymentDatesAdjustments.'

   <payRelativeTo> PayRelativeToEnum </payRelativeTo> [1]
    'Specifies whether the payments occur relative to each adjusted calculation period start date, adjusted calculation period end date or each reset
    date. The reset date is applicable in the case of certain euro (former French Franc) floating rate indices. Calculation period start date means
    relative to the start of the first calculation period contributing to a given payment. Similarly, calculation period end date means the end of the
    last calculation period contributing to a given payment.The valuation date is applicable for Brazilian-CDI swaps.'

   <paymentDaysOffset> Offset </paymentDaysOffset> [0..1]
    'If early payment or delayed payment is required, specifies the number of days offset that the payment occurs relative to what would otherwise be the
    unadjusted payment date. The offset can be specified in terms of either calendar or business days. Even in the case of a calendar days offset, the
    resulting payment date, adjusted for the specified calendar days offset, will still be adjusted in accordance with the specified payment dates
    adjustments. This element should only be included if early or delayed payment is applicable, i.e. if the periodMultiplier element value is not equal to
    zero. An early payment would be indicated by a negative periodMultiplier element value and a delayed payment (or payment lag) would be indicated by a
    positive periodMultiplier element value.'

   <paymentDatesAdjustments> BusinessDayAdjustments </paymentDatesAdjustments> [1]
    'The business day convention to apply to each payment date if it would otherwise fall on a day that is not a business day in the specified financial
    business centers.'

</...>
```

Diagram


PaymentDates
A type defining parameters used to generate the payment dates schedule, including the specification of early or delayed payments. Payment dates are determined relative to the calculation period dates or the reset dates.
calculationPeriodDatesReference
A pointer style reference to the associated calculation period dates component defined elsewhere in the document.
CalculationPeriodDatesReference
type: CalculationPeriodDatesReference
resetDatesReference
A pointer style reference to the associated reset dates component defined elsewhere in the document.
ResetDatesReference
type: ResetDatesReference
valuationDatesReference
A pointer style reference to the associated valuation dates component defined elsewhere in the document. Implemented for Brazilian-CDI Swaps where it will refer to the settlementProvision/nonDeliverableSettlement/fxFixingDate structure.
ValuationDatesReference
type: ValuationDatesReference
paymentFrequency
The frequency at which regular payment dates occur. If the payment frequency is equal to the frequency defined in the calculation period dates component then one calculation period contributes to each payment amount. If the payment frequency is less frequent than the frequency defined in the calculation period dates component then more than one calculation period will contribute to the payment amount. A payment frequency more frequent than the calculation period frequency or one that is not a multiple of the calculation period frequency is invalid. If the payment frequency is of value T (term), the period is defined by the swap\swapStream\calculationPerioDates\effectiveDate and the swap\swapStream\calculationPerioDates\terminationDate.
Frequency
type: Frequency from: fpml-shared-4-8.xsd
firstPaymentDate
0..1
The first unadjusted payment date. This day may be subject to adjustment in accordance with any business day convention specified in paymentDatesAdjustments. This element must only be included if there is an initial stub. This date will normally correspond to an unadjusted calculation period start or end date. This is true even if early or delayed payment is specified to be applicable since the actual first payment date will be the specified number of days before or after the applicable adjusted calculation period start or end date with the resulting payment date then being adjusted in accordance with any business day convention specified in paymentDatesAdjustments.
date
type: date from: XMLSchema.xsd
lastRegularPaymentDate
0..1
The last regular unadjusted payment date. This day may be subject to adjustment in accordance with any business day convention specified in paymentDatesAdjustments. This element must only be included if there is a final stub. All calculation periods after this date contribute to the final payment. The final payment is made relative to the final set of calculation periods or the final reset date as the case may be. This date will normally correspond to an unadjusted calculation period start or end date. This is true even if early or delayed payment is specified to be applicable since the actual last regular payment date will be the specified number of days before or after the applicable adjusted calculation period start or end date with the resulting payment date then being adjusted in accordance with any business day convention specified in paymentDatesAdjustments.
date
type: date from: XMLSchema.xsd
payRelativeTo
Specifies whether the payments occur relative to each adjusted calculation period start date, adjusted calculation period end date or each reset date. The reset date is applicable in the case of certain euro (former French Franc) floating rate indices. Calculation period start date means relative to the start of the first calculation period contributing to a given payment. Similarly, calculation period end date means the end of the last calculation period contributing to a given payment.The valuation date is applicable for Brazilian-CDI swaps.



Schema Component Representation

```
<xsd:complexType name="PaymentDates">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="calculationPeriodDatesReference" type=" CalculationPeriodDatesReference "/>
         <xsd:element name="resetDatesReference" type=" ResetDatesReference "/>
         <xsd:element name="valuationDatesReference" type=" ValuationDatesReference "/>
      </xsd:choice>
      <xsd:element name="paymentFrequency" type=" Frequency "/>
      <xsd:element name="firstPaymentDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="lastRegularPaymentDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="payRelativeTo" type=" PayRelativeToEnum "/>
      <xsd:element name="paymentDaysOffset" type=" Offset " minOccurs="0"/>
      <xsd:element name="paymentDatesAdjustments" type=" BusinessDayAdjustments "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentDatesReference

[Table of contents]

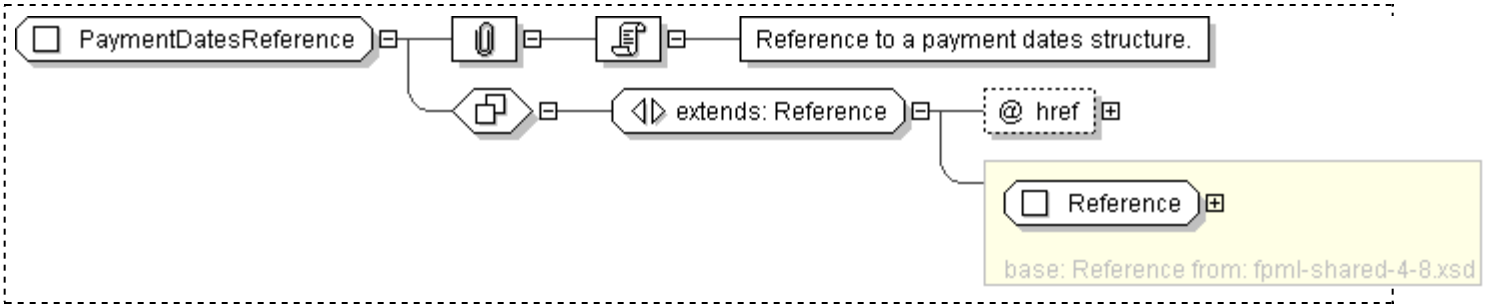
Super-types:	Reference < PaymentDatesReference (by extension)
Sub-types:	None

Name	PaymentDatesReference
Used by (from the same schema document)	Complex Type DateRelativeToPaymentDates
Abstract	no
Documentation	Reference to a payment dates structure.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentDatesReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="PaymentDates"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PriceSourceDisruption

[Table of contents]

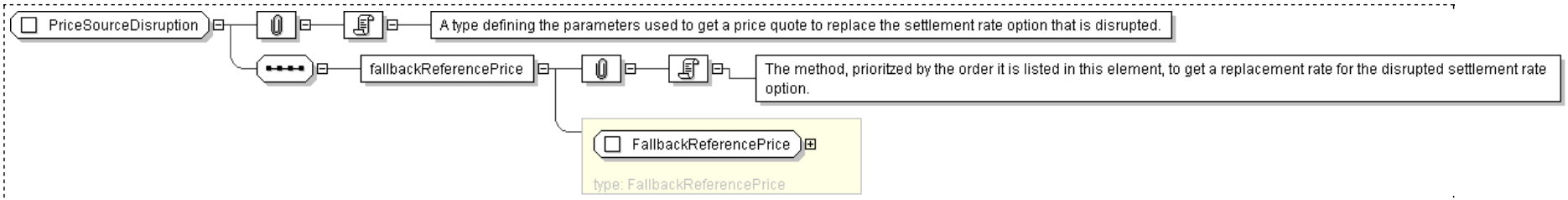
Super-types:	None
Sub-types:	None

Name	PriceSourceDisruption
Used by (from the same schema document)	Complex Type NonDeliverableSettlement
Abstract	no
Documentation	A type defining the parameters used to get a price quote to replace the settlement rate option that is disrupted.

XML Instance Representation

```
<...>
  <fallbackReferencePrice> FallbackReferencePrice </fallbackReferencePrice> [1]
  'The method, prioritzed by the order it is listed in this element, to get a replacement rate for the disrupted settlement rate option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PriceSourceDisruption">
  <xsd:sequence>
    <xsd:element name="fallbackReferencePrice" type=" FallbackReferencePrice "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrincipalExchange

[Table of contents]

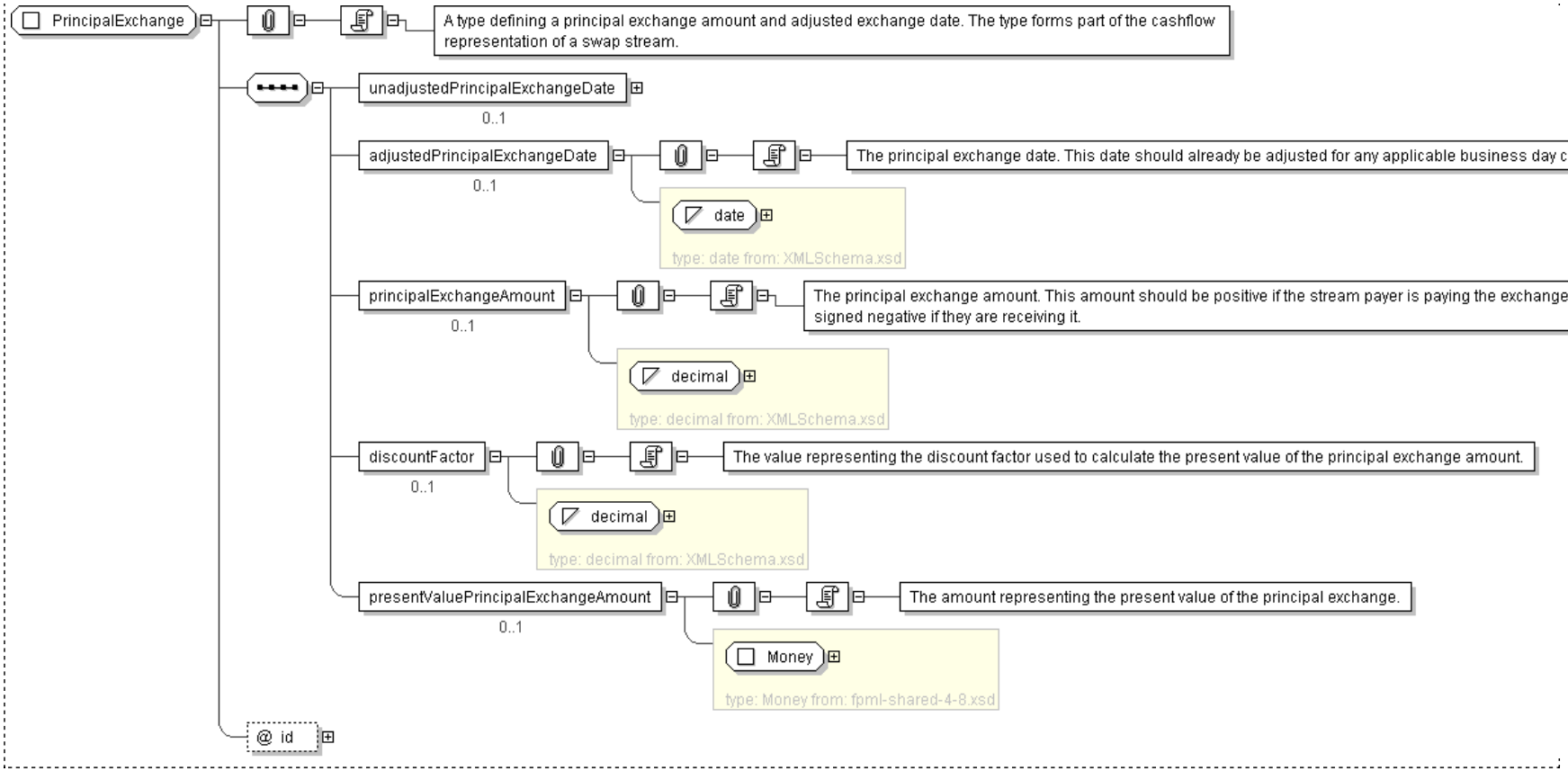
Super-types:	None
Sub-types:	None

Name	PrincipalExchange
Used by (from the same schema document)	Complex Type Cashflows
Abstract	no
Documentation	A type defining a principal exchange amount and adjusted exchange date. The type forms part of the cashflow representation of a swap stream.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedPrincipalExchangeDate> xsd:date </unadjustedPrincipalExchangeDate> [0..1]
   <adjustedPrincipalExchangeDate> xsd:date </adjustedPrincipalExchangeDate> [0..1]
   'The principal exchange date. This date should already be adjusted for any applicable business day convention.'

   <principalExchangeAmount> xsd:decimal </principalExchangeAmount> [0..1]
   'The principal exchange amount. This amount should be positive if the stream payer is paying the exchange amount and signed negative if they
   are receiving it.'

   <discountFactor> xsd:decimal </discountFactor> [0..1]
   'The value representing the discount factor used to calculate the present value of the principal exchange amount.'

   <presentValuePrincipalExchangeAmount> Money </presentValuePrincipalExchangeAmount> [0..1]
   'The amount representing the present value of the principal exchange.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchange">
  <xsd:sequence>
    <xsd:element name="unadjustedPrincipalExchangeDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="adjustedPrincipalExchangeDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="principalExchangeAmount" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="discountFactor" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="presentValuePrincipalExchangeAmount" type=" Money " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RelevantUnderlyingDateReference

[Table of contents]

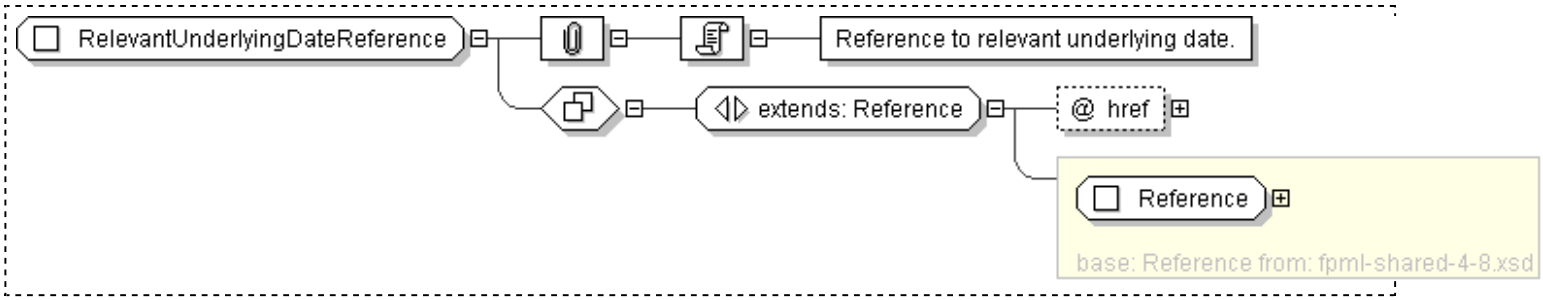
Super-types:	Reference < RelevantUnderlyingDateReference (by extension)
Sub-types:	None

Name	RelevantUnderlyingDateReference
Used by (from the same schema document)	Complex Type FinalCalculationPeriodDateAdjustment
Abstract	no
Documentation	Reference to relevant underlying date.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RelevantUnderlyingDateReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResetDates

[Table of contents]

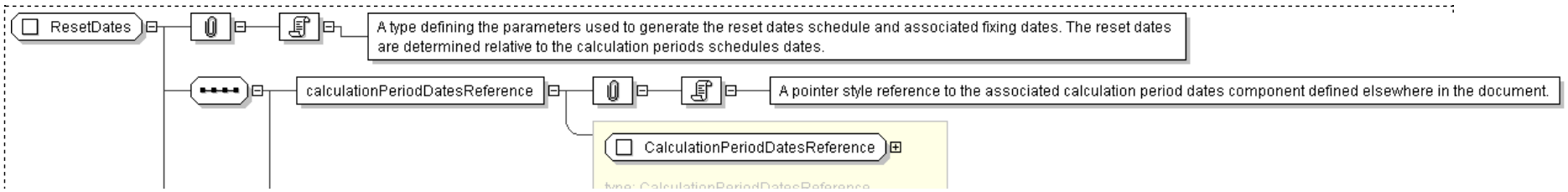
Super-types:	None
Sub-types:	None

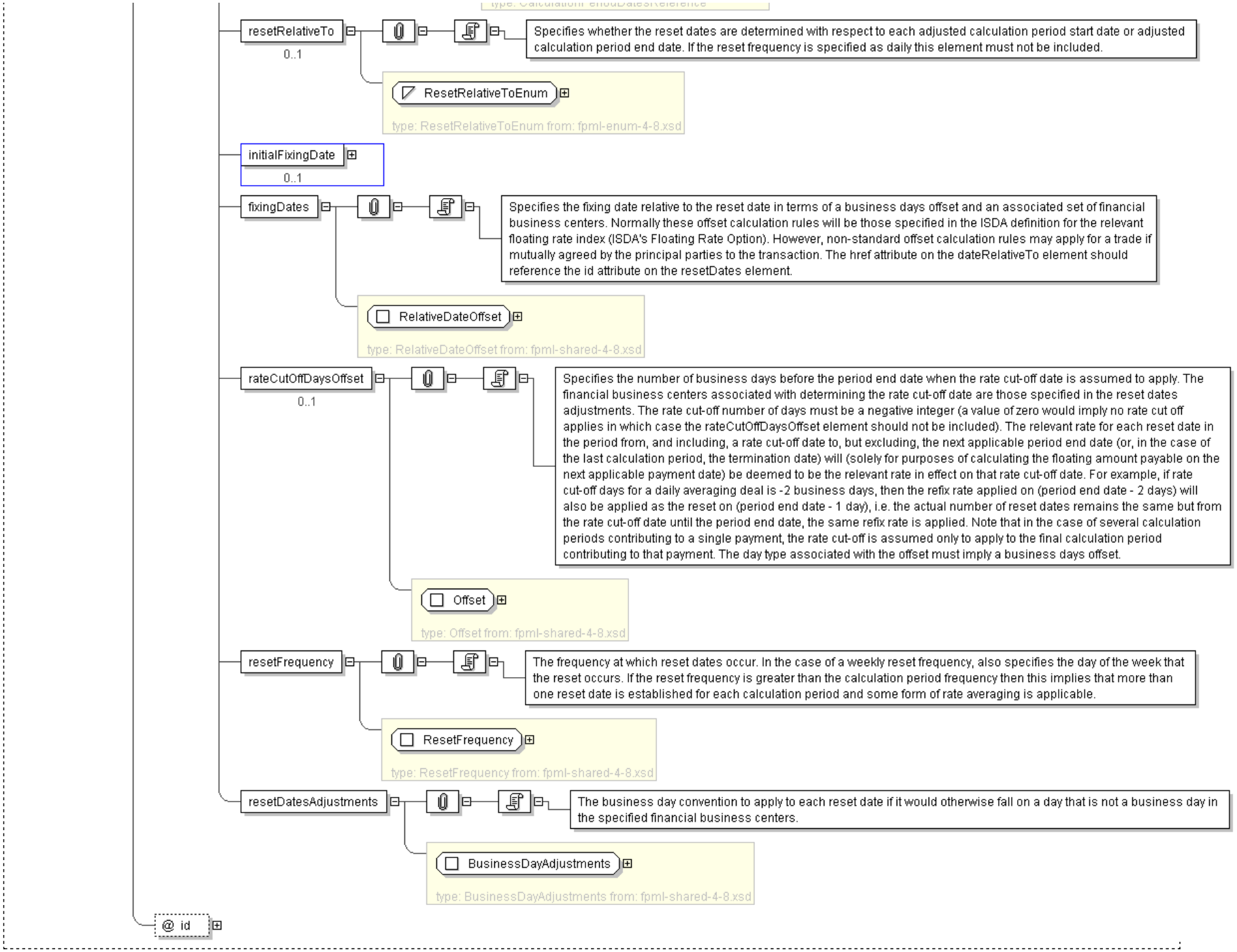
Name	ResetDates
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining the parameters used to generate the reset dates schedule and associated fixing dates. The reset dates are determined relative to the calculation periods schedules dates.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
   <calculationPeriodDatesReference> CalculationPeriodDatesReference </calculationPeriodDatesReference> [1]
   'A pointer style reference to the associated calculation period dates component defined elsewhere in the document.'

   <resetRelativeTo> ResetRelativeToEnum </resetRelativeTo> [0..1]
   'Specifies whether the reset dates are determined with respect to each adjusted calculation period start date or adjusted calculation period
   end date. If the reset frequency is specified as daily this element must not be included.'

   <initialFixingDate> RelativeDateOffset </initialFixingDate> [0..1]
   <fixingDates> RelativeDateOffset </fixingDates> [1]
   'Specifies the fixing date relative to the reset date in terms of a business days offset and an associated set of financial business
   centers. Normally these offset calculation rules will be those specified in the ISDA definition for the relevant floating rate index
   (ISDA\'s Floating Rate Option). However, non-standard offset calculation rules may apply for a trade if mutually agreed by the principal
   parties to the transaction. The href attribute on the dateRelativeTo element should reference the id attribute on the resetDates element.'

   <rateCutOffDaysOffset> Offset </rateCutOffDaysOffset> [0..1]
   'Specifies the number of business days before the period end date when the rate cut-off date is assumed to apply. The financial business
   centers associated with determining the rate cut-off date are those specified in the reset dates adjustments. The rate cut-off number of
   days must be a negative integer (a value of zero would imply no rate cut off applies in which case the rateCutOffDaysOffset element should
   not be included). The relevant rate for each reset date in the period from, and including, a rate cut-off date to, but excluding, the next
   applicable period end date (or, in the case of the last calculation period, the termination date) will (solely for purposes of calculating
   the floating amount payable on the next applicable payment date) be deemed to be the relevant rate in effect on that rate cut-off date. For
   example, if rate cut-off days for a daily averaging deal is -2 business days, then the refix rate applied on (period end date - 2 days) will
   also be applied as the reset on (period end date - 1 day), i.e. the actual number of reset dates remains the same but from the rate cut-off
   date until the period end date, the same refix rate is applied. Note that in the case of several calculation periods contributing to a
   single payment, the rate cut-off is assumed only to apply to the final calculation period contributing to that payment. The day type
   associated with the offset must imply a business days offset.'

   <resetFrequency> ResetFrequency </resetFrequency> [1]
   'The frequency at which reset dates occur. In the case of a weekly reset frequency, also specifies the day of the week that the reset
   occurs. If the reset frequency is greater than the calculation period frequency then this implies that more than one reset date is
   established for each calculation period and some form of rate averaging is applicable.'

   <resetDatesAdjustments> BusinessDayAdjustments </resetDatesAdjustments> [1]
   'The business day convention to apply to each reset date if it would otherwise fall on a day that is not a business day in the specified
   financial business centers.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="ResetDates">
  <xsd:sequence>
    <xsd:element name="calculationPeriodDatesReference" type=" CalculationPeriodDatesReference "/>
    <xsd:element name="resetRelativeTo" type=" ResetRelativeToEnum " minOccurs="0"/>
```

```
      <xsd:element name="initialFixingDate" type=" RelativeDateOffset " minOccurs="0"/>
      <xsd:element name="fixingDates" type=" RelativeDateOffset "/>
      <xsd:element name="rateCutOffDaysOffset" type=" Offset " minOccurs="0"/>
      <xsd:element name="resetFrequency" type=" ResetFrequency "/>
      <xsd:element name="resetDatesAdjustments" type=" BusinessDayAdjustments "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResetDatesReference

[Table of contents]

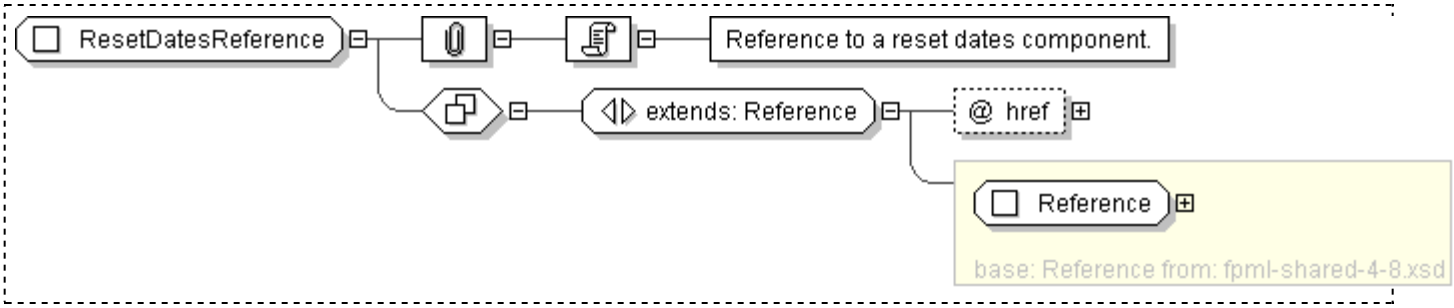
Super-types:	Reference < ResetDatesReference (by extension)
Sub-types:	None

Name	ResetDatesReference
Used by (from the same schema document)	Complex Type PaymentDates
Abstract	no
Documentation	Reference to a reset dates component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResetDatesReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="ResetDates"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementProvision

[Table of contents]

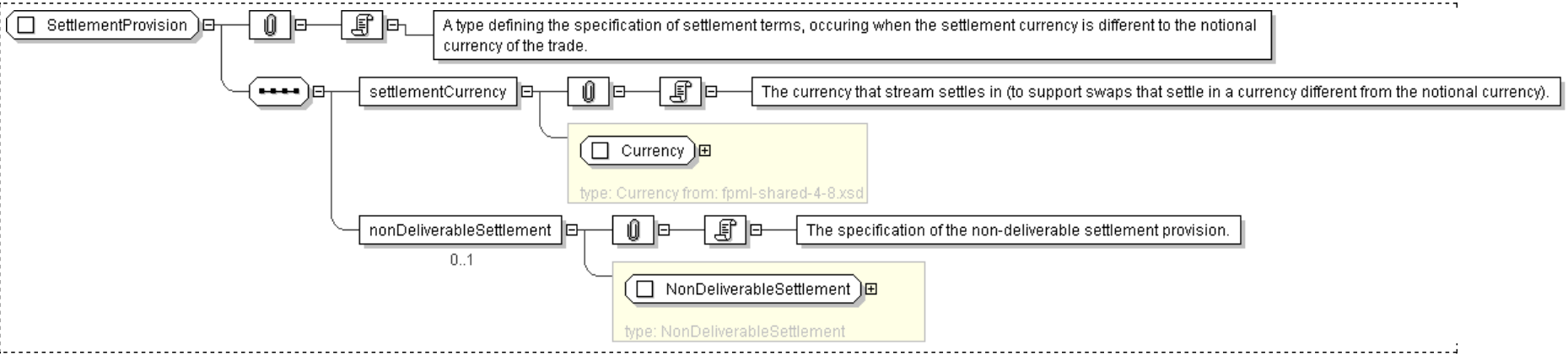
Super-types:	None
Sub-types:	None

Name	SettlementProvision
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining the specification of settlement terms, occuring when the settlement currency is different to the notional currency of the trade.

XML Instance Representation

```
<...>
  <settlementCurrency> Currency </settlementCurrency> [1]
  'The currency that stream settles in (to support swaps that settle in a currency different from the notional currency).'

  <nonDeliverableSettlement> NonDeliverableSettlement </nonDeliverableSettlement> [0..1]
  'The specification of the non-deliverable settlement provision.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementProvision">
  <xsd:sequence>
    <xsd:element name="settlementCurrency" type=" Currency "/>
    <xsd:element name="nonDeliverableSettlement" type=" NonDeliverableSettlement " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementRateOption

[Table of contents]

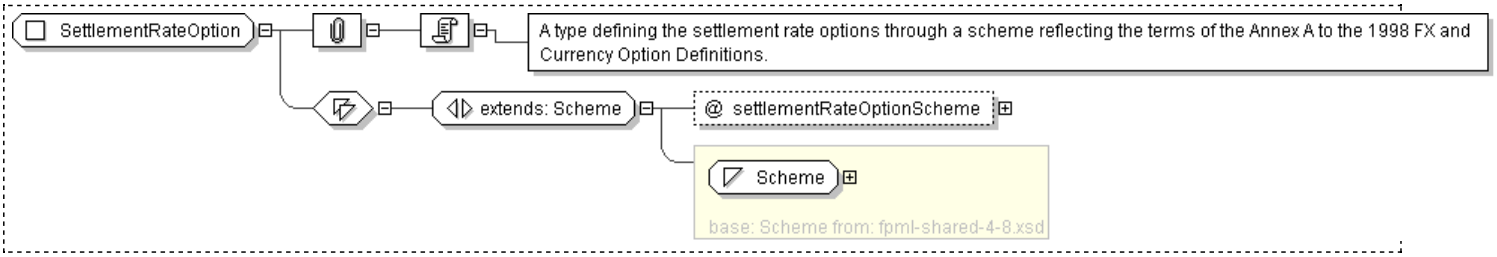
Super-types:	Scheme < SettlementRateOption (by extension)
Sub-types:	None

Name	SettlementRateOption
Used by (from the same schema document)	Complex Type FallbackReferencePrice , Complex Type NonDeliverableSettlement
Abstract	no
Documentation	A type defining the settlement rate options through a scheme reflecting the terms of the Annex A to the 1998 FX and Currency Option Definitions.

XML Instance Representation

```
<...
 settlementRateOptionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementRateOption">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="settlementRateOptionScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/settlement-rate-option"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SinglePartyOption

[Table of contents]

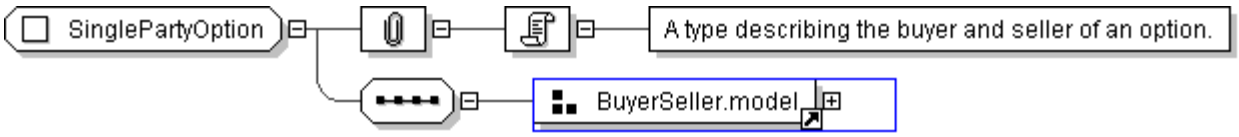
Super-types:	None
Sub-types:	None

Name	SinglePartyOption
Used by (from the same schema document)	Complex Type OptionalEarlyTermination
Abstract	no
Documentation	A type describing the buyer and seller of an option.

XML Instance Representation

```
<...>
   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this
    instrument and receives the rights defined by it. See 2000 ISDA definitions
    Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that
    grants the rights defined by this instrument and in return receives a payment
    for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is
    the floating rate payer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SinglePartyOption">
   <xsd:sequence>
      <xsd:group ref=" BuyerSeller.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StubCalculationPeriodAmount

[Table of contents]

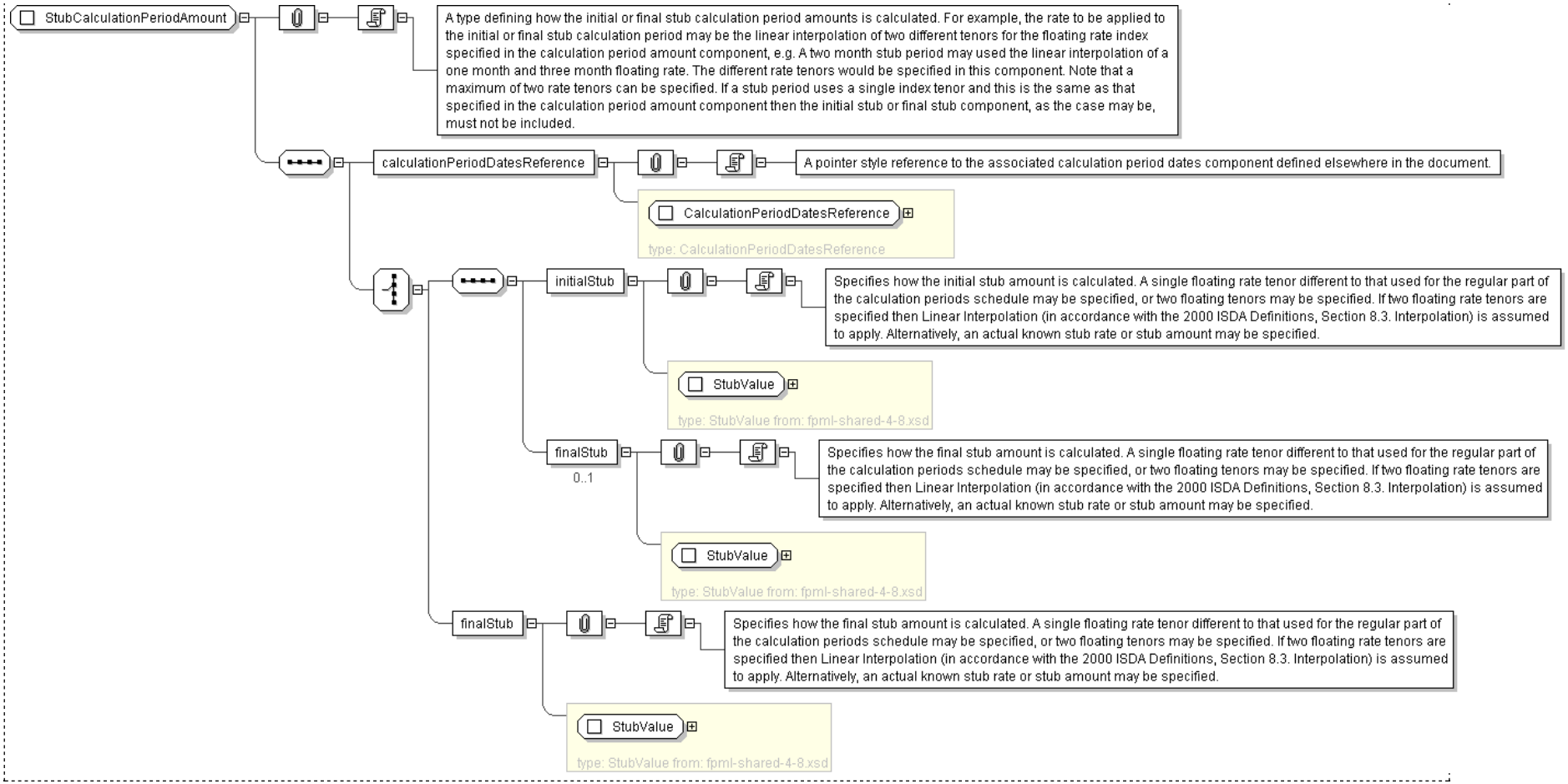
Super-types:	None
Sub-types:	None

Name	StubCalculationPeriodAmount
Used by (from the same schema document)	Complex Type InterestRateStream
Abstract	no
Documentation	A type defining how the initial or final stub calculation period amounts is calculated. For example, the rate to be applied to the initial or final stub calculation period may be the linear interpolation of two different tenors for the floating rate index specified in the calculation period amount component, e.g. A two month stub period may used the linear interpolation of a one month and three month floating rate. The different rate tenors would be specified in this component. Note that a maximum of two rate tenors can be specified. If a stub period uses a single index tenor and this is the same as that specified in the calculation period amount component then the initial stub or final stub component, as the case may be, must not be included.

XML Instance Representation

```
<...>
   <calculationPeriodDatesReference> CalculationPeriodDatesReference </calculationPeriodDatesReference> [1]
   'A pointer style reference to the associated calculation period dates component defined elsewhere in the document.'

   Start Choice [1]
      <initialStub> StubValue </initialStub> [1]
      'Specifies how the initial stub amount is calculated. A single floating rate tenor different to that used for the regular part of the calculation periods
      schedule may be specified, or two floating tenors may be specified. If two floating rate tenors are specified then Linear Interpolation (in accordance with
      the 2000 ISDA Definitions, Section 8.3. Interpolation) is assumed to apply. Alternatively, an actual known stub rate or stub amount may be specified.'

      <finalStub> StubValue </finalStub> [0..1]
      'Specifies how the final stub amount is calculated. A single floating rate tenor different to that used for the regular part of the calculation periods
      schedule may be specified, or two floating tenors may be specified. If two floating rate tenors are specified then Linear Interpolation (in accordance with
      the 2000 ISDA Definitions, Section 8.3. Interpolation) is assumed to apply. Alternatively, an actual known stub rate or stub amount may be specified.'

      <finalStub> StubValue </finalStub> [1]
      'Specifies how the final stub amount is calculated. A single floating rate tenor different to that used for the regular part of the calculation periods
      schedule may be specified, or two floating tenors may be specified. If two floating rate tenors are specified then Linear Interpolation (in accordance with
      the 2000 ISDA Definitions, Section 8.3. Interpolation) is assumed to apply. Alternatively, an actual known stub rate or stub amount may be specified.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StubCalculationPeriodAmount">
  <xsd:sequence>
    <xsd:element name="calculationPeriodDatesReference" type=" CalculationPeriodDatesReference "/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="initialStub" type=" StubValue "/>
        <xsd:element name="finalStub" type=" StubValue " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="finalStub" type=" StubValue "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Swap

[Table of contents]

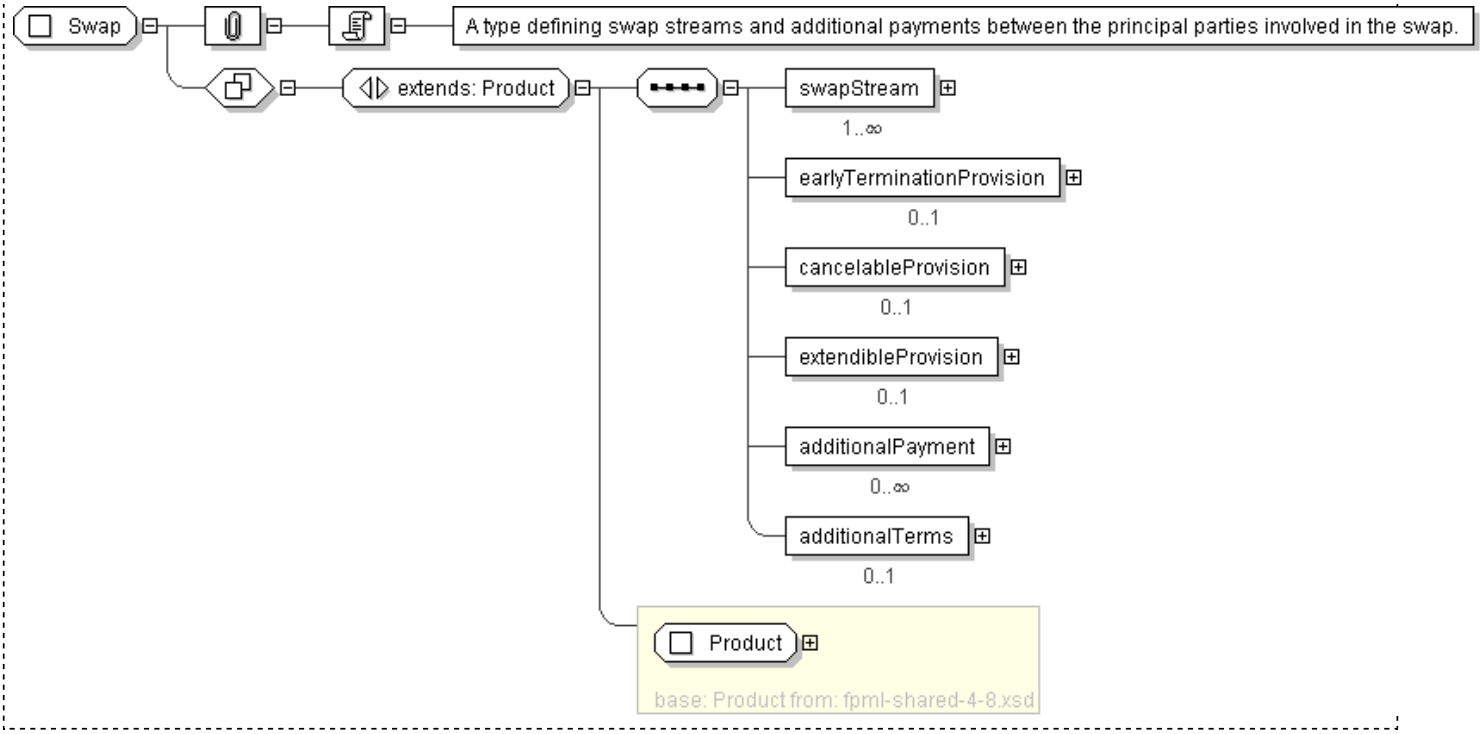
Super-types:	Product < Swap (by extension)
Sub-types:	None

Name	Swap
Used by (from the same schema document)	Element swap
Abstract	no
Documentation	A type defining swap streams and additional payments between the principal parties involved in the swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a
    coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values
    associated with this element. Note that the domain values for this element are not strictly an
    enumerated list.'

   <swapStream> InterestRateStream </swapStream> [1..*]
    'The swap streams.'

   <earlyTerminationProvision> EarlyTerminationProvision </earlyTerminationProvision> [0..1]
    'Parameters specifying provisions relating to the optional and mandatory early terminarion of
    a swap transaction.'

   <cancelableProvision> CancelableProvision </cancelableProvision> [0..1]
    'A provision that allows the specification of an embedded option within a swap giving the
    buyer of the option the right to terminate the swap, in whole or in part, on the early
    termination date.'

   <extendibleProvision> ExtendibleProvision </extendibleProvision> [0..1]
    'A provision that allows the specification of an embedded option with a swap giving the buyer
    of the option the right to extend the swap, in whole or in part, to the extended termination
    date.'

   <additionalPayment> Payment </additionalPayment> [0..*]
    'Additional payments between the principal parties.'

   <additionalTerms> SwapAdditionalTerms </additionalTerms> [0..1]
    'Contains any additional terms to the swap contract.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Swap">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="swapStream" type=" InterestRateStream " maxOccurs="unbounded"/>
        <xsd:element name="earlyTerminationProvision" type=" EarlyTerminationProvision "
        minOccurs="0"/>
        <xsd:element name="cancelableProvision" type=" CancelableProvision " minOccurs="0"/>
        <xsd:element name="extendibleProvision" type=" ExtendibleProvision " minOccurs="0"/>
        <xsd:element name="additionalPayment" type=" Payment " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="additionalTerms" type=" SwapAdditionalTerms " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SwapAdditionalTerms

[Table of contents]

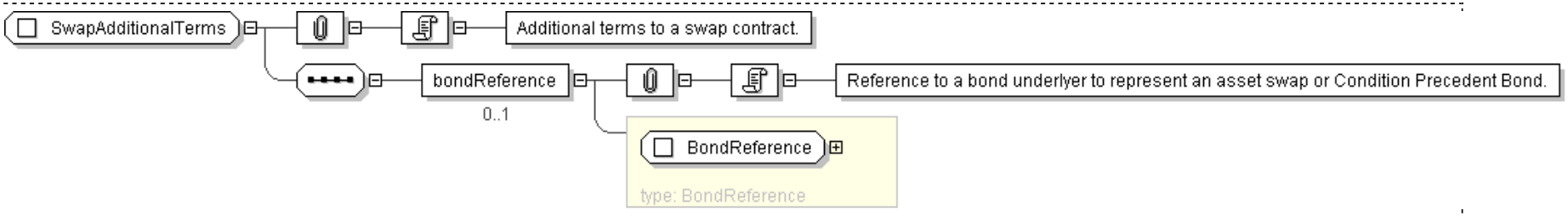
Super-types:	None
Sub-types:	None

Name	SwapAdditionalTerms
Used by (from the same schema document)	Complex Type Swap
Abstract	no
Documentation	Additional terms to a swap contract.

XML Instance Representation

```
<...>
   <bondReference> BondReference </bondReference> [0..1]
   'Reference to a bond underlyer to represent an asset swap or Condition Precedent Bond.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SwapAdditionalTerms">
   <xsd:sequence>
      <xsd:element name="bondReference" type=" BondReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Swaption

[Table of contents]

Super-types:	Product < Swaption (by extension)
Sub-types:	None

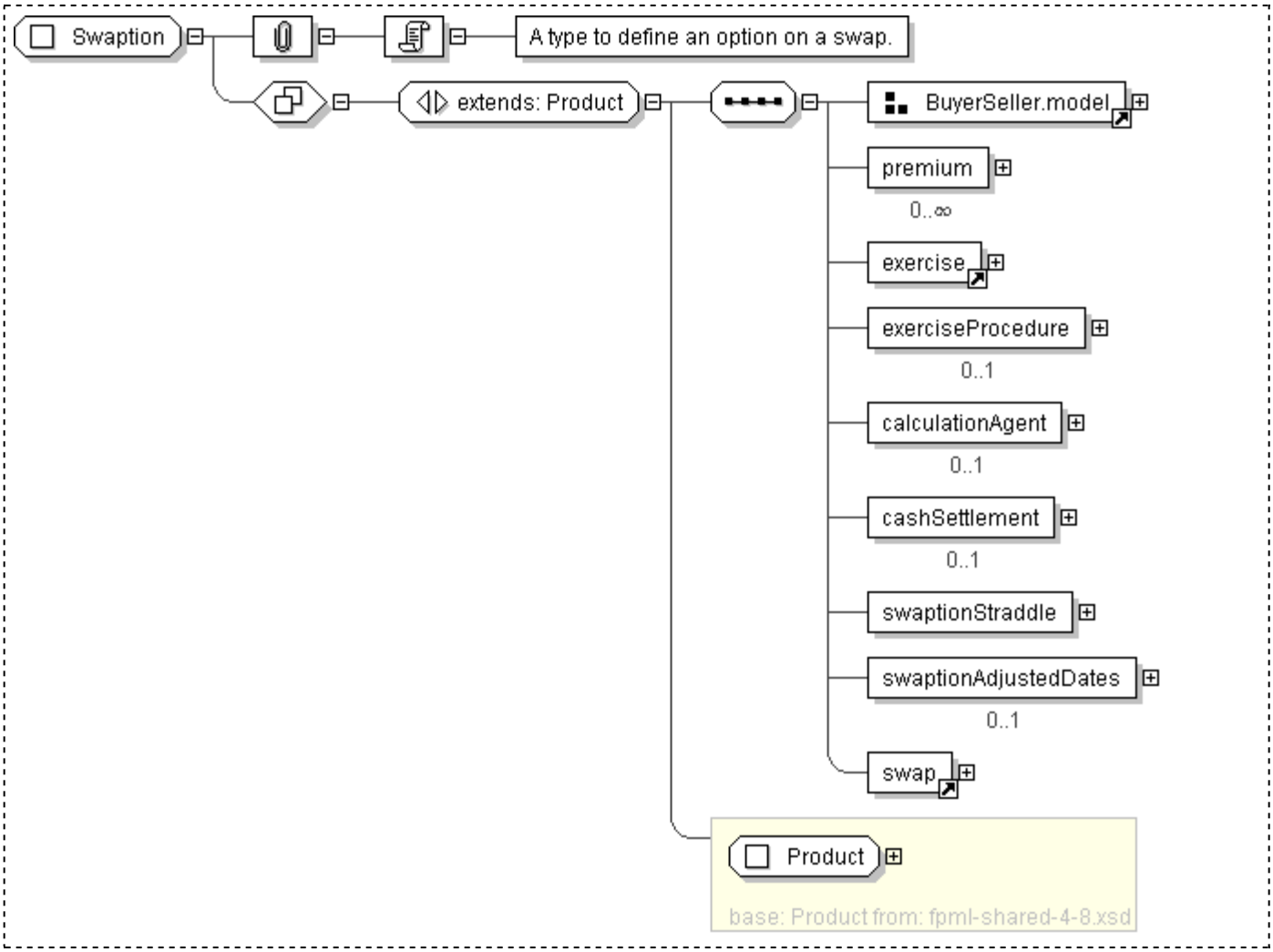
Name	Swaption
Used by (from the same schema document)	Element swaption
Abstract	no
Documentation	A type to define an option on a swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this
    instrument and receives the rights defined by it. See 2000 ISDA definitions
    Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that
    grants the rights defined by this instrument and in return receives a payment
    for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is
    the floating rate payer.'

   <premium> Payment </premium> [0..*]
    'The option premium amount payable by buyer to seller on the specified payment
    date.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [0..1]
    'A set of parameters defining procedures associated with the exercise.'

   <calculationAgent> CalculationAgent </calculationAgent> [0..1]
    'The ISDA Calculation Agent responsible for performing duties associated with
    an optional early termination.'

   <cashSettlement> CashSettlement </cashSettlement> [0..1]
    'If specified, this means that cash settlement is applicable to the transaction
    and defines the parameters associated with the cash settlement prodcedure. If
    not specified, then physical settlement is applicable.'

   <swaptionStraddle> xsd:boolean </swaptionStraddle> [1]
    'Whether the option is a swaption or a swaption straddle.'

   <swaptionAdjustedDates> SwaptionAdjustedDates </swaptionAdjustedDates> [0..1]
    'The adjusted dates associated with swaption exercise. These dates have been
```

```
   adjusted for any applicable business day convention.'

   <swap> ... </swap> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Swaption">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:group ref=" BuyerSeller.model "/>
            <xsd:element name="premium" type=" Payment " minOccurs="0"
            maxOccurs="unbounded"/>
            <xsd:element ref=" exercise "/>
            <xsd:element name="exerciseProcedure" type=" ExerciseProcedure "
            minOccurs="0"/>
            <xsd:element name="calculationAgent" type=" CalculationAgent "
            minOccurs="0"/>
            <xsd:element name="cashSettlement" type=" CashSettlement " minOccurs="0"/>
            <xsd:element name="swaptionStraddle" type=" xsd:boolean "/>
            <xsd:element name="swaptionAdjustedDates" type=" SwaptionAdjustedDates "
            minOccurs="0"/>
            <xsd:element ref=" swap "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SwaptionAdjustedDates

[Table of contents]

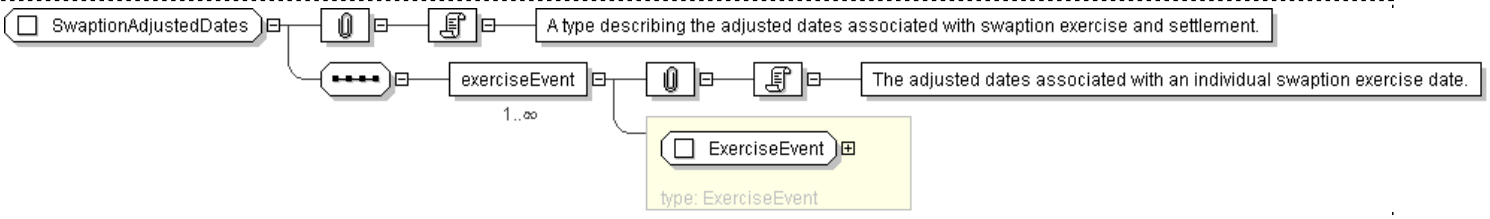
Super-types:	None
Sub-types:	None

Name	SwaptionAdjustedDates
Used by (from the same schema document)	Complex Type Swaption
Abstract	no
Documentation	A type describing the adjusted dates associated with swaption exercise and settlement.

XML Instance Representation

```
<...>
  <exerciseEvent> ExerciseEvent </exerciseEvent> [1..*]
  'The adjusted dates associated with an individual swaption exercise date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SwaptionAdjustedDates">
  <xsd:sequence>
    <xsd:element name="exerciseEvent" type=" ExerciseEvent " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationDatesReference

[Table of contents]

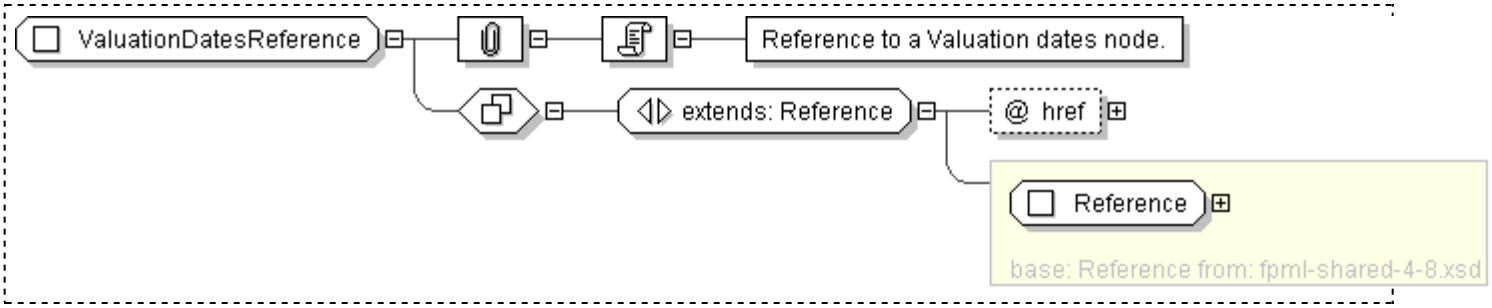
Super-types:	Reference < ValuationDatesReference (by extension)
Sub-types:	None

Name	ValuationDatesReference
Used by (from the same schema document)	Complex Type PaymentDates
Abstract	no
Documentation	Reference to a Valuation dates node.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationDatesReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="FxFixingDate"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationPostponement

[Table of contents]

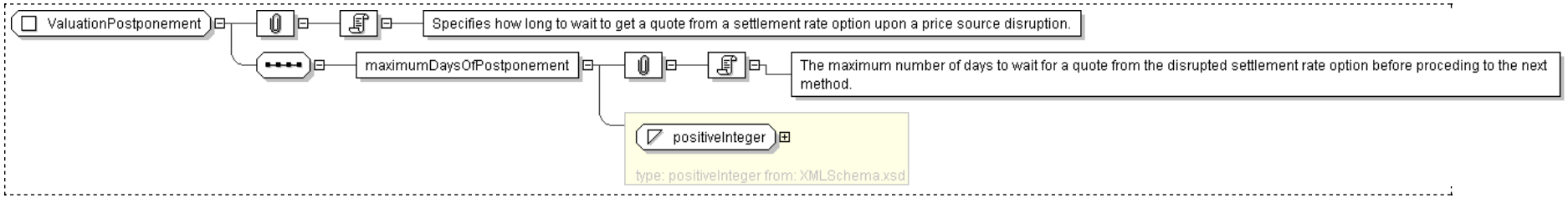
Super-types:	None
Sub-types:	None

Name	ValuationPostponement
Used by (from the same schema document)	Complex Type FallbackReferencePrice
Abstract	no
Documentation	Specifies how long to wait to get a quote from a settlement rate option upon a price source disruption.

XML Instance Representation

```
<...>
  <maximumDaysOfPostponement> xsd:positiveInteger </maximumDaysOfPostponement> [1]
   'The maximum number of days to wait for a quote from the disrupted settlement rate option before proceding to the next method.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationPostponement">
  <xsd:sequence>
    <xsd:element name="maximumDaysOfPostponement" type=" xsd:positiveInteger "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: YieldCurveMethod

[Table of contents]

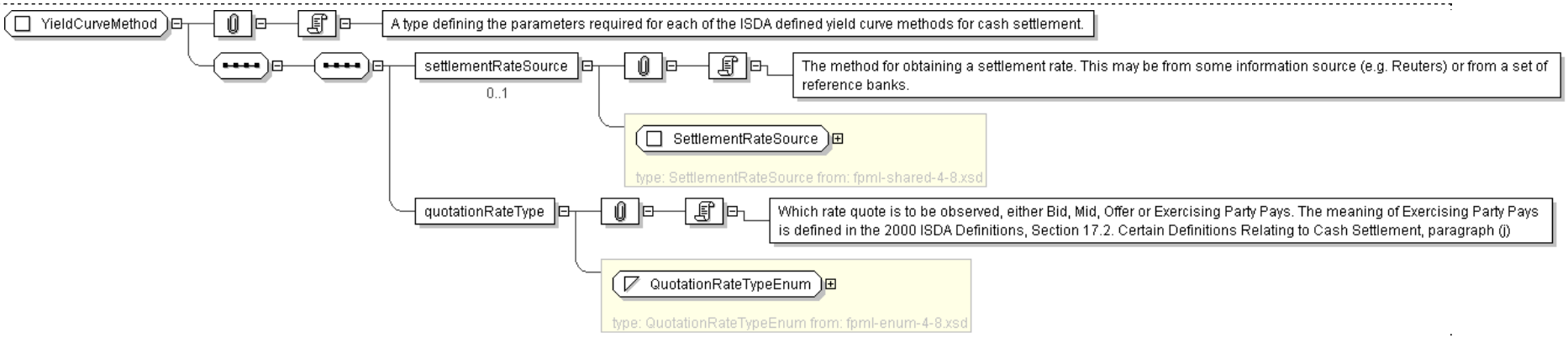
Super-types:	None
Sub-types:	None

Name	YieldCurveMethod
Used by (from the same schema document)	Complex Type CashSettlement , Complex Type CashSettlement , Complex Type CashSettlement
Abstract	no
Documentation	A type defining the parameters required for each of the ISDA defined yield curve methods for cash settlement.

XML Instance Representation

```
<...>
   <settlementRateSource> SettlementRateSource </settlementRateSource> [0..1]
    'The method for obtaining a settlement rate. This may be from some information source (e.g. Reuters) or from a set of reference banks.'

   <quotationRateType> QuotationRateTypeEnum </quotationRateType> [1]
    'Which rate quote is to be observed, either Bid, Mid, Offer or Exercising Party Pays. The meaning of Exercising Party Pays is defined in the 2000
    ISDA Definitions, Section 17.2. Certain Definitions Relating to Cash Settlement, paragraph (j)'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="YieldCurveMethod">
  <xsd:sequence>
    <xsd:sequence>
      <xsd:element name="settlementRateSource" type=" SettlementRateSource " minOccurs="0"/>
      <xsd:element name="quotationRateType" type=" QuotationRateTypeEnum "/>
    </xsd:sequence>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: BorrowingOptionType
  - Complex Type: DealSummary
  - Complex Type: DrawdownNotice
  - Complex Type: DrawdownPayment
  - Complex Type: FacilityCommitmentPosition
  - Complex Type: FacilityNotice
  - Complex Type: FacilityRepayment
  - Complex Type: FacilitySummary
  - Complex Type: FeeAccrualPeriod
  - Complex Type: FeeAccrualSchedule
  - Complex Type: FxTerms
  - Complex Type: FxTermsSchedule
  - Complex Type: InterestAccrualPeriod
  - Complex Type: InterestAccrualSchedule
  - Complex Type: InterestPayment
  - Complex Type: InterestPaymentNotice
  - Complex Type: InterestRatePeriod
  - Complex Type: LcAmendmentNotice
  - Complex Type: LcBalanceNotice
  - Complex Type: LcEvergreenOption
  - Complex Type: LcIssuanceNotice
  - Complex Type: LcPosition
  - Complex Type: LcSummary
  - Complex Type: LcTerminationNotice
  - Complex Type: LenderLoanContractPeriod
  - Complex Type: LenderPositionPeriod
  - Complex Type: LetterOfCredit
  - Complex Type: LoanContract
  - Complex Type: LoanContractNotice
  - Complex Type: LoanContractPosition
  - Complex Type: LoanContractRepayment
  - Complex Type: LoanContractSummary
  - Complex Type: MarginRateChange
  - Complex Type: MaturingLoanContract
  - Complex Type: MaturingLoanContracts
  - Complex Type: NewLoanContracts
  - Complex Type: OnGoingFeeNotice
  - Complex Type: OnGoingFeePayment
  - Complex Type: OnGoingFeeRateChange
  - Complex Type: OneOffFeeNotice
  - Complex Type: OneOffFeePayment
  - Complex Type: ParticipationAmount
  - Complex Type: PikPeriod
  - Complex Type: PricingChangeNotice
  - Complex Type: PricingChangeReason
  - Complex Type: RatePeriod
  - Complex Type: Repayment
  - Complex Type: RepaymentNotice
  - Complex Type: RolloverNotice
  - Model Group: FacilityNoticeDetails.model
  - Model Group: InterestPaymentDetails.model
- Legend

- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 2406 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ◦ fpml-asset-4-8.xsd
	  ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 2406 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-asset-4-8.xsd"/>
  <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress
Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress

Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See:

http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: FacilityNoticeDetails.model

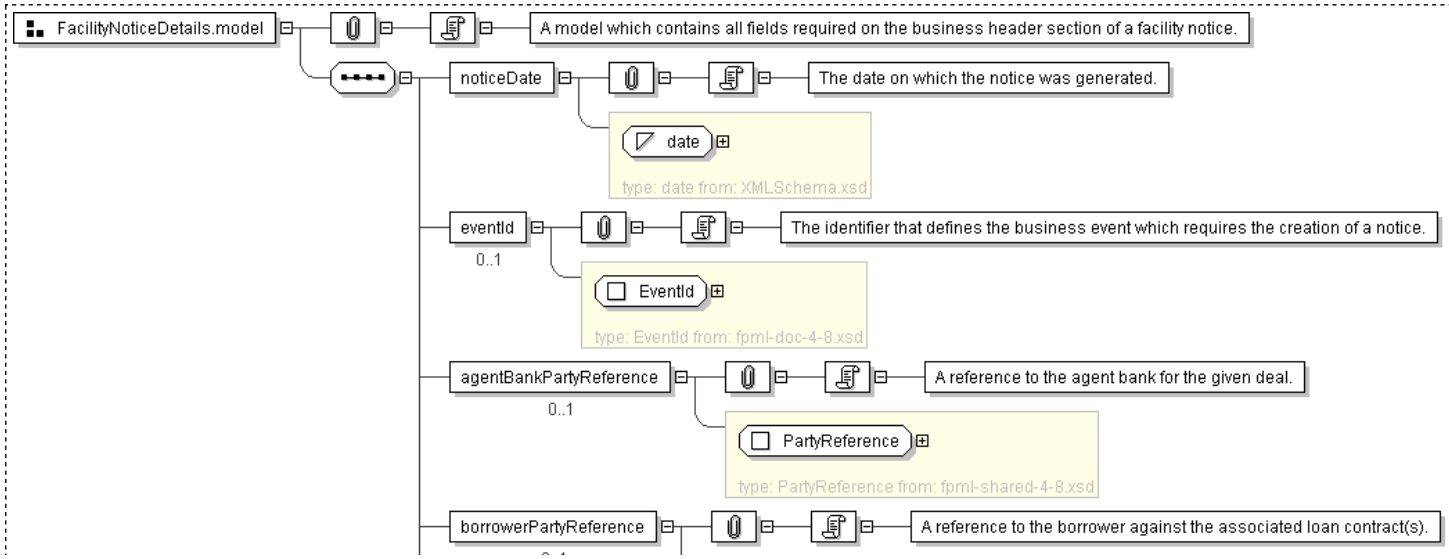
[Table of contents]

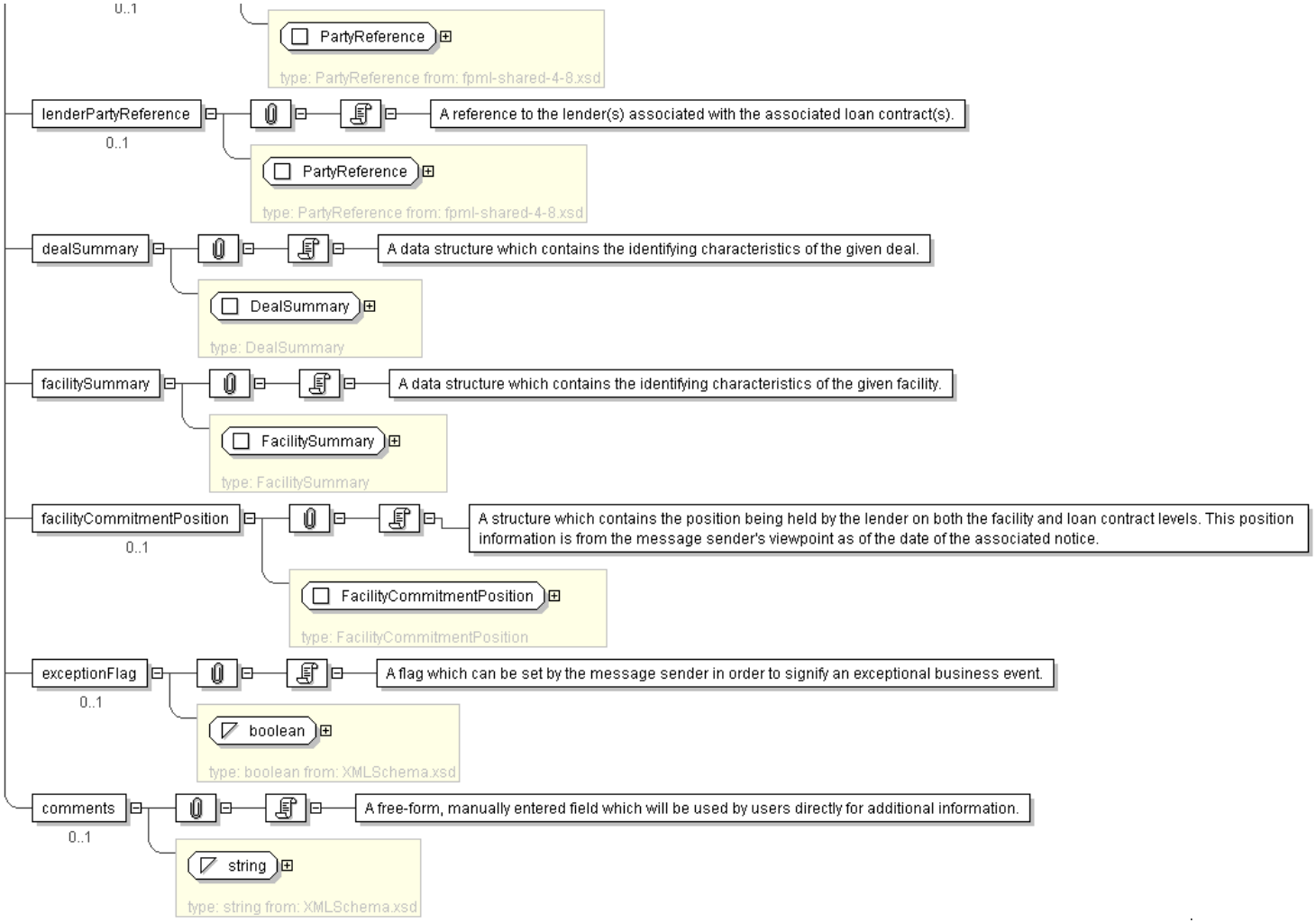
Name	FacilityNoticeDetails.model
Used by (from the same schema document)	Complex Type FacilityNotice , Complex Type LoanContractNotice
Documentation	A model which contains all fields required on the business header section of a facility notice.

XML Instance Representation

```
<noticeDate> xsd:date </noticeDate> [1]
'The date on which the notice was generated.'

<eventId> EventId </eventId> [0..1]
'The identifier that defines the business event which requires the creation of a notice.'

<agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
'A reference to the agent bank for the given deal.'

<borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
'A reference to the borrower against the associated loan contract(s).'

<lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
'A reference to the lender(s) associated with the associated loan contract(s).'

<dealSummary> DealSummary </dealSummary> [1]
'A data structure which contains the identifying characteristics of the given deal.'

<facilitySummary> FacilitySummary </facilitySummary> [1]
'A data structure which contains the identifying characteristics of the given facility.'

<facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is from the
message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
'A flag which can be set by the message sender in order to signify an exceptional business event.'

<comments> xsd:string </comments> [0..1]
'A free-form, manually entered field which will be used by users directly for additional information.'
```

Diagram





Schema Component Representation

```
<xsd:group name="FacilityNoticeDetails.model">
   <xsd:sequence>
      <xsd:element name="noticeDate" type=" xsd:date "/>
      <xsd:element name="eventId" type=" EventId " minOccurs="0"/>
      <xsd:element name="agentBankPartyReference" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="borrowerPartyReference" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="lenderPartyReference" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="dealSummary" type=" DealSummary "/>
      <xsd:element name="facilitySummary" type=" FacilitySummary "/>
      <xsd:element name="facilityCommitmentPosition" type=" FacilityCommitmentPosition " minOccurs="0"/>
      <xsd:element name="exceptionFlag" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="comments" type=" xsd:string " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: InterestPaymentDetails.model

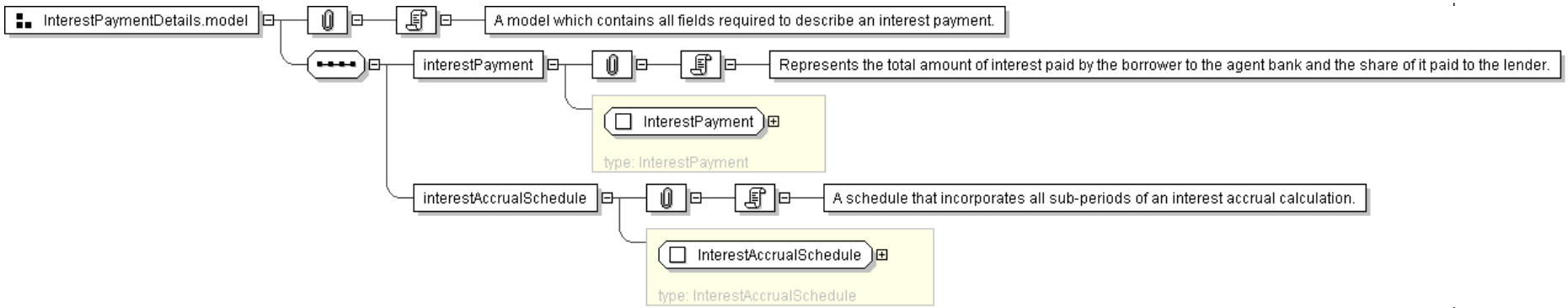
[Table of contents]

Name	InterestPaymentDetails.model
Used by (from the same schema document)	Complex Type InterestPaymentNotice , Complex Type MaturingLoanContract
Documentation	A model which contains all fields required to describe an interest payment.

XML Instance Representation

```
<interestPayment> InterestPayment </interestPayment> [1]
'Represents the total amount of interest paid by the borrower to the agent bank and the share of it paid to the lender.'

<interestAccrualSchedule> InterestAccrualSchedule </interestAccrualSchedule> [1]
'A schedule that incorporates all sub-periods of an interest accrual calculation.'
```

Diagram



Schema Component Representation

```
<xsd:group name="InterestPaymentDetails.model">
  <xsd:sequence>
    <xsd:element name="interestPayment" type=" InterestPayment "/>
    <xsd:element name="interestAccrualSchedule" type=" InterestAccrualSchedule "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BorrowingOptionType

[Table of contents]

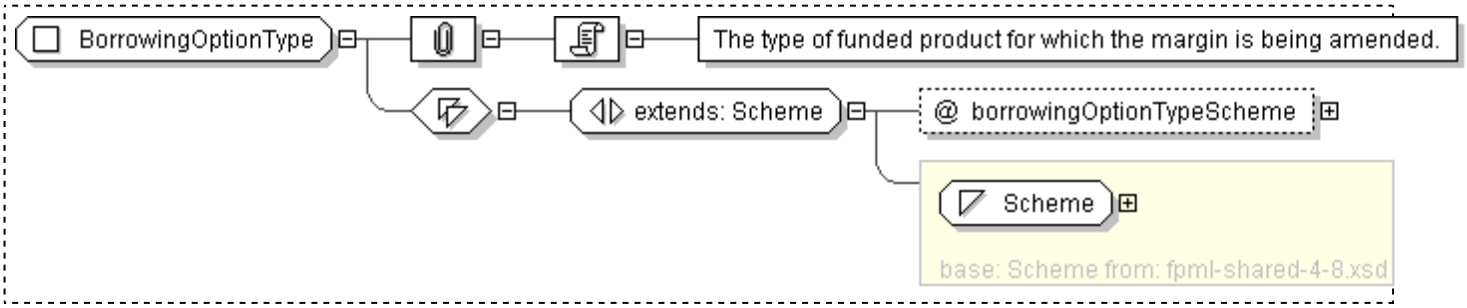
Super-types:	Scheme < BorrowingOptionType (by extension)
Sub-types:	None

Name	BorrowingOptionType
Used by (from the same schema document)	Complex Type MarginRateChange
Abstract	no
Documentation	The type of funded product for which the margin is being amended.

XML Instance Representation

```
<...
 borrowingOptionTypeScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BorrowingOptionType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="borrowingOptionTypeScheme" type=" xsd:anyURI "
         use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DealSummary

[Table of contents]

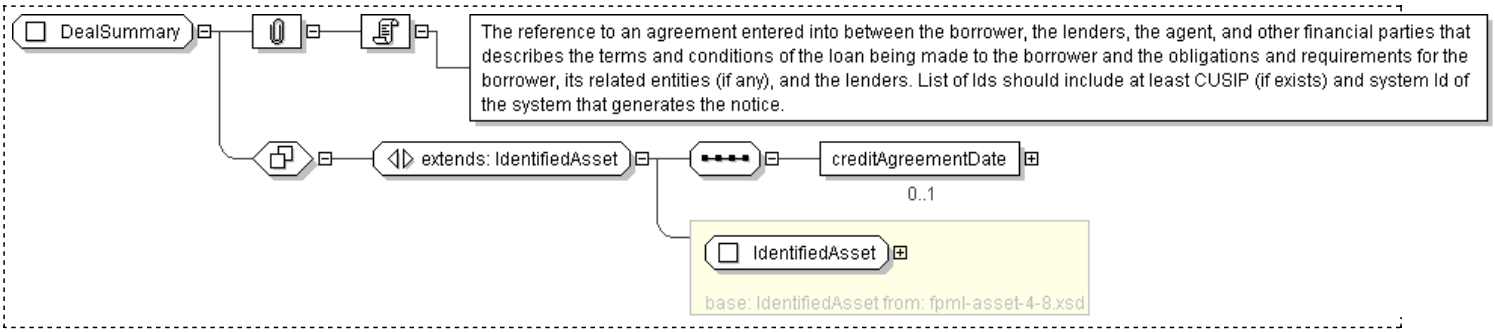
Super-types:	IdentifiedAsset < DealSummary (by extension)
Sub-types:	None

Name	DealSummary
Used by (from the same schema document)	Model Group FacilityNoticeDetails.model
Abstract	no
Documentation	The reference to an agreement entered into between the borrower, the lenders, the agent, and other financial parties that describes the terms and conditions of the loan being made to the borrower and the obligations and requirements for the borrower, its related entities (if any), and the lenders. List of Ids should include at least CUSIP (if exists) and system Id of the system that generates the notice.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
     'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
     'Long name of the underlying asset.'

   <creditAgreementDate> xsd:date </creditAgreementDate> [0..1]
     'The credit agreement date is the closing date (the date where the agreement has been signed) for the loans
     in the credit agreement. Funding of the facilities occurs on (or sometimes a little after) the Credit
     Agreement date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DealSummary">
  <xsd:complexContent>
    <xsd:extension base=" IdentifiedAsset ">
      <xsd:sequence>
        <xsd:element name="creditAgreementDate" type=" xsd:date " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DrawdownNotice

[Table of contents]

Super-types:	NotificationMessage < LoanContractNotice (by extension) < DrawdownNotice (by extension)
Sub-types:	None

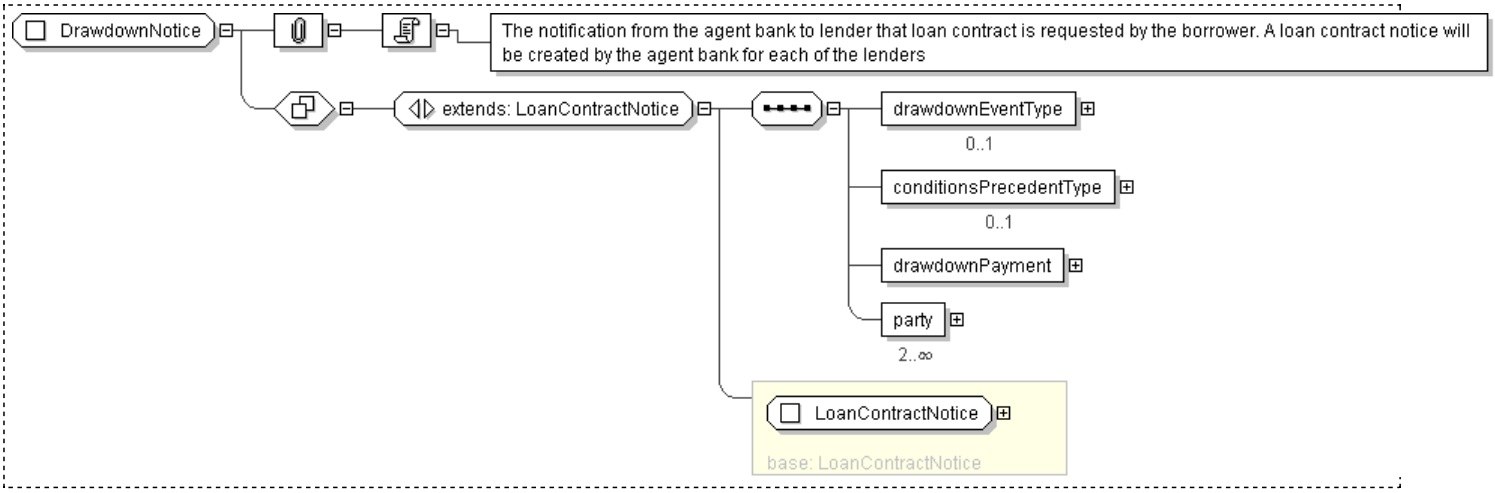
Name	DrawdownNotice
Abstract	no
Documentation	The notification from the agent bank to lender that loan contract is requested by the borrower. A loan contract notice will be created by the agent bank for each of the lenders

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan contract
   levels. This position information is from the message sender\'s viewpoint as of the date of the associated
   notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional information.'

   Start Choice [1]
'The sender may choose to either transmit the full or partial loan contract details.'


      <loanContract> LoanContract </loanContract> [1]
      'A core structure describing a loan contract between borrower and lenders forming part or all of the
      credit line offered by a facility structure within a deal.'

      <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
      'A basic set of fields used to uniquely identify the loan contract.'

   End Choice
   <drawdownEventType> DrawdownEventTypeEnum </drawdownEventType> [0..1]
   'An enumeration that describes whether this message is a drawdown or a rate set notification. The same
   message structure is used for both, with some business validation differences.'

   <conditionsPrecedentType> ConditionsPrecedentEnum </conditionsPrecedentType> [0..1]
   'An enumeration which describes whether the condition precent have been met, not met or been waived. Please
```

```
   note: this field in not required since conditions precedent may not be applicable in certain scenarios.'

   <drawdownPayment> DrawdownPayment </drawdownPayment> [1]
   'The lender\'s portion of the drawdown payment.'

   <party> Party </party> [2..*]
   'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DrawdownNotice">
  <xsd:complexContent>
    <xsd:extension base=" LoanContractNotice ">
      <xsd:sequence>
        <xsd:element name="drawdownEventType" type=" DrawdownEventTypeEnum " minOccurs="0"/>
        <xsd:element name="conditionsPrecedentType" type=" ConditionsPrecedentEnum " minOccurs="0"/>
        <xsd:element name="drawdownPayment" type=" DrawdownPayment "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DrawdownPayment

[Table of contents]

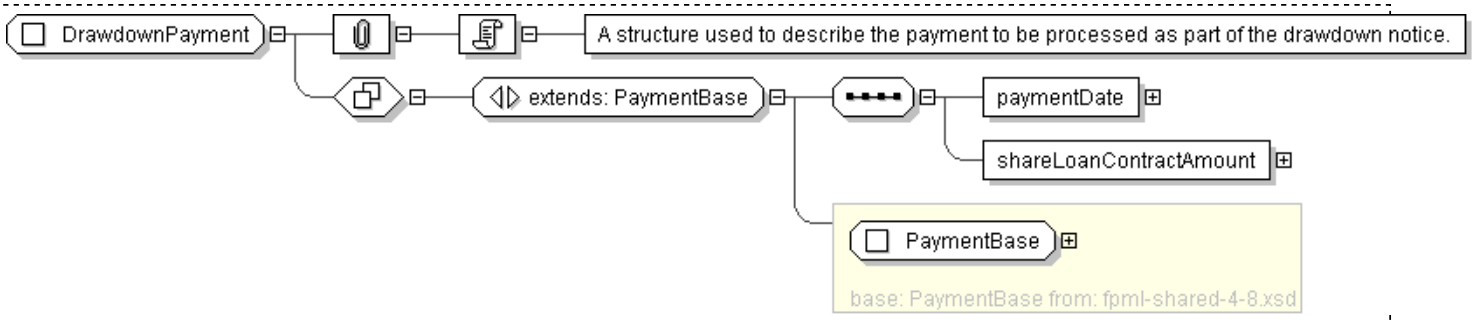
Super-types:	PaymentBase < DrawdownPayment (by extension)
Sub-types:	None

Name	DrawdownPayment
Used by (from the same schema document)	Complex Type DrawdownNotice , Complex Type NewLoanContracts
Abstract	no
Documentation	A structure used to describe the payment to be processed as part of the drawdown notice.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <paymentDate> xsd:date </paymentDate> [1]
   'The date on which the principal payment is made by the lender to the agent bank. This is an
   actual (adjusted) date. Usually defaults to the effective date of the loan contract.'

   <shareLoanContractAmount> Money </shareLoanContractAmount> [1]
   'The amount of principal allocated to the lender for the given drawdown request. This is the
   same as the payment that must be made by the lender to satisfy the drawdown request.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DrawdownPayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="paymentDate" type=" xsd:date "/>
        <xsd:element name="shareLoanContractAmount" type=" Money "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FacilityCommitmentPosition

[Table of contents]

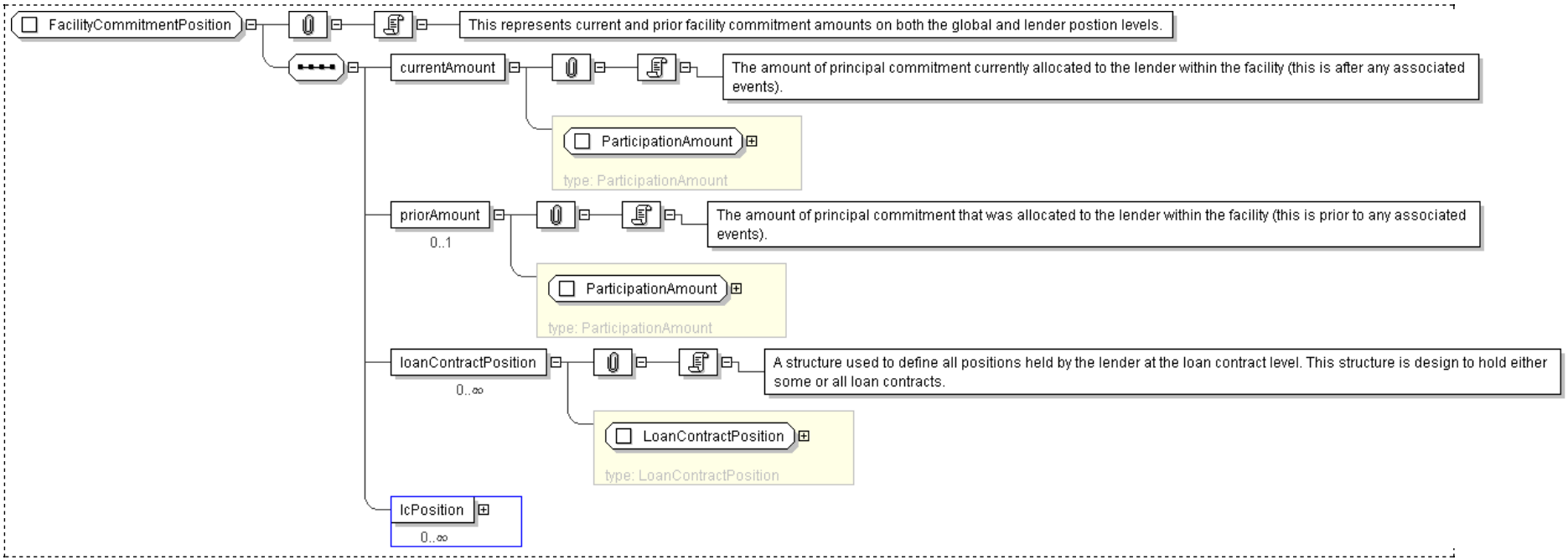
Super-types:	None
Sub-types:	None

Name	FacilityCommitmentPosition
Used by (from the same schema document)	Model Group FacilityNoticeDetails.model
Abstract	no
Documentation	This represents current and prior facility commitment amounts on both the global and lender postion levels.

XML Instance Representation

```
<...>
   <currentAmount> ParticipationAmount </currentAmount> [1]
    'The amount of principal commitment currently allocated to the lender within the facility (this is after any associated events).'

   <priorAmount> ParticipationAmount </priorAmount> [0..1]
    'The amount of principal commitment that was allocated to the lender within the facility (this is prior to any associated events).'

   <loanContractPosition> LoanContractPosition </loanContractPosition> [0..*]
    'A structure used to define all positions held by the lender at the loan contract level. This structure is design to hold either some or all loan
    contracts.'

   <lcPosition> LcPosition </lcPosition> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FacilityCommitmentPosition">
```

```
  <xsd:sequence>
    <xsd:element name="currentAmount" type=" ParticipationAmount "/>
    <xsd:element name="priorAmount" type=" ParticipationAmount " minOccurs="0"/>
    <xsd:element name="loanContractPosition" type=" LoanContractPosition " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="lcPosition" type=" LcPosition " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FacilityNotice

[Table of contents]

Super-types: NotificationMessage < FacilityNotice (by extension)

Sub-types:

- LcAmendmentNotice (by extension)
- LcBalanceNotice (by extension)
- LcIssuanceNotice (by extension)
- LcTerminationNotice (by extension)
- OneOffFeeNotice (by extension)
- OnGoingFeeNotice (by extension)
- PricingChangeNotice (by extension)
- RepaymentNotice (by extension)
- RolloverNotice (by extension)

Name	FacilityNotice
Abstract	yes
Documentation	A base type for all facility-level notices.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'
```

<dealSummary> DealSummary **</dealSummary>** [1]

'A data structure which contains the identifying characteristics of the given deal.'

<facilitySummary> FacilitySummary **</facilitySummary>** [1]

'A data structure which contains the identifying characteristics of the given facility.'

<facilityCommitmentPosition> FacilityCommitmentPosition **</facilityCommitmentPosition>** [0..1]

'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean **</exceptionFlag>** [0..1]

'A flag which can be set by the message sender in order to signify an exceptional business event.'

<comments> xsd:string **</comments>** [0..1]

'A free-form, manually entered field which will be used by users directly for additional information.'

</...>

Diagram

Schema Component Representation

```
<xsd:complexType name="FacilityNotice" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:group ref=" FacilityNoticeDetails.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FacilityRepayment

[Table of contents]

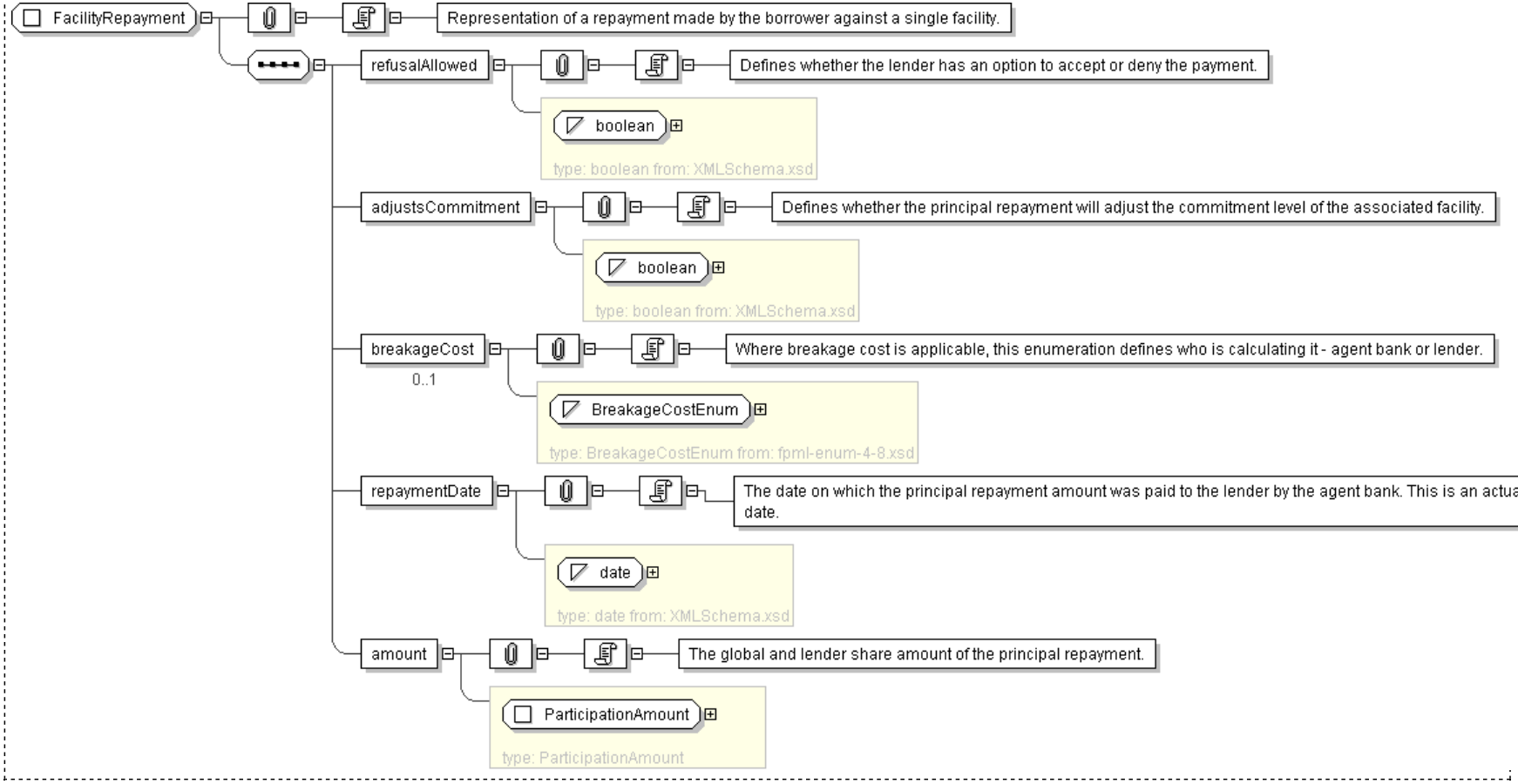
Super-types:	None
Sub-types:	None

Name	FacilityRepayment
Used by (from the same schema document)	Complex Type Repayment
Abstract	no
Documentation	Representation of a repayment made by the borrower against a single facility.

XML Instance Representation

```
<...>
  <refusalAllowed> xsd:boolean </refusalAllowed> [1]
  'Defines whether the lender has an option to accept or deny the payment.'

  <adjustsCommitment> xsd:boolean </adjustsCommitment> [1]
  'Defines whether the principal repayment will adjust the commitment level of the associated facility.'

  <breakageCost> BreakageCostEnum </breakageCost> [0..1]
  'Where breakage cost is applicable, this enumeration defines who is calculating it - agent bank or lender.'

  <repaymentDate> xsd:date </repaymentDate> [1]
  'The date on which the principal repayment amount was paid to the lender by the agent bank. This is an actual (adjusted) date.'

  <amount> ParticipationAmount </amount> [1]
  'The global and lender share amount of the principal repayment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FacilityRepayment">
  <xsd:sequence>
    <xsd:element name="refusalAllowed" type=" xsd:boolean "/>
    <xsd:element name="adjustsCommitment" type=" xsd:boolean "/>
    <xsd:element name="breakageCost" type=" BreakageCostEnum " minOccurs="0"/>
    <xsd:element name="repaymentDate" type=" xsd:date "/>
    <xsd:element name="amount" type=" ParticipationAmount "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FacilitySummary

[Table of contents]

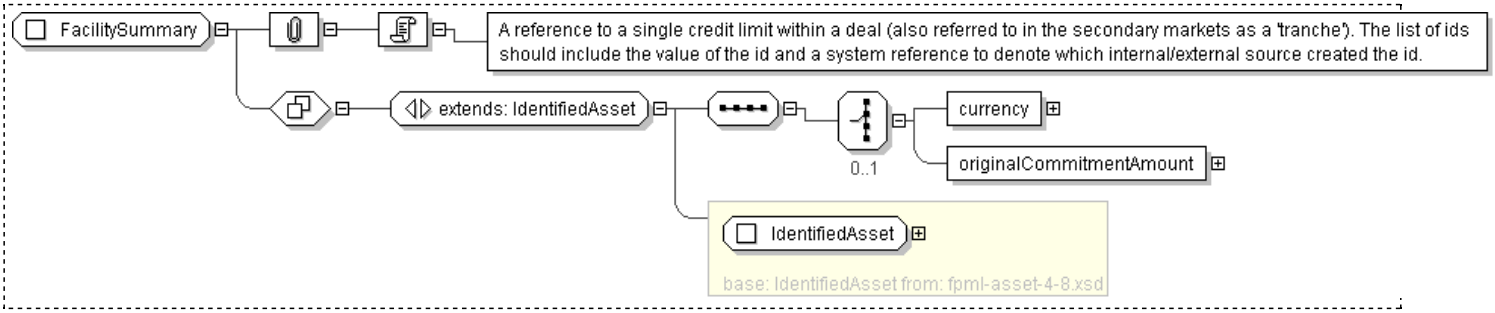
Super-types:	IdentifiedAsset < FacilitySummary (by extension)
Sub-types:	None

Name	FacilitySummary
Used by (from the same schema document)	Model Group FacilityNoticeDetails.model
Abstract	no
Documentation	A reference to a single credit limit within a deal (also referred to in the secondary markets as a 'tranche'). The list of ids should include the value of the id and a system reference to denote which internal/external source created the id.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <instrumentId> InstrumentId </instrumentId> [1..*]
   'Identification of the underlying asset, using public and/or private identifiers.'

   <description> xsd:string </description> [0..1]
   'Long name of the underlying asset.'

   Start Choice [0..1]
      <currency> Currency </currency> [1]
      'Facility denomination currency.'

      <originalCommitmentAmount> Money </originalCommitmentAmount> [1]
      'Original global commitment amount of the facility.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FacilitySummary">
   <xsd:complexContent>
      <xsd:extension base=" IdentifiedAsset ">
         <xsd:sequence>
            <xsd:choice minOccurs="0">
               <xsd:element name="currency" type=" Currency "/>
               <xsd:element name="originalCommitmentAmount" type=" Money "/>
            </xsd:choice>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FeeAccrualPeriod

[Table of contents]

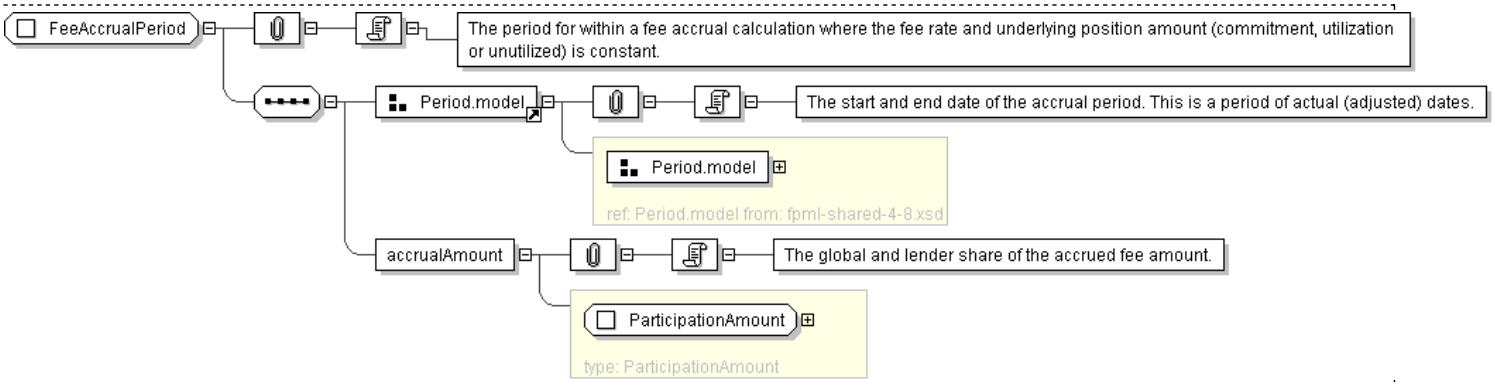
Super-types:	None
Sub-types:	None

Name	FeeAccrualPeriod
Used by (from the same schema document)	Complex Type FeeAccrualSchedule
Abstract	no
Documentation	The period for within a fee accrual calculation where the fee rate and underlying position amount (commitment, utilization or unutilized) is constant.

XML Instance Representation

```
<...>
  <startDate> xsd:date </startDate> [1]
  'Date on which this period begins.'

  <endDate> xsd:date </endDate> [1]
  'Date on which this period ends.'

  <accrualAmount> ParticipationAmount </accrualAmount> [1]
  'The global and lender share of the accrued fee amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FeeAccrualPeriod">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:element name="accrualAmount" type=" ParticipationAmount "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FeeAccrualSchedule

[Table of contents]

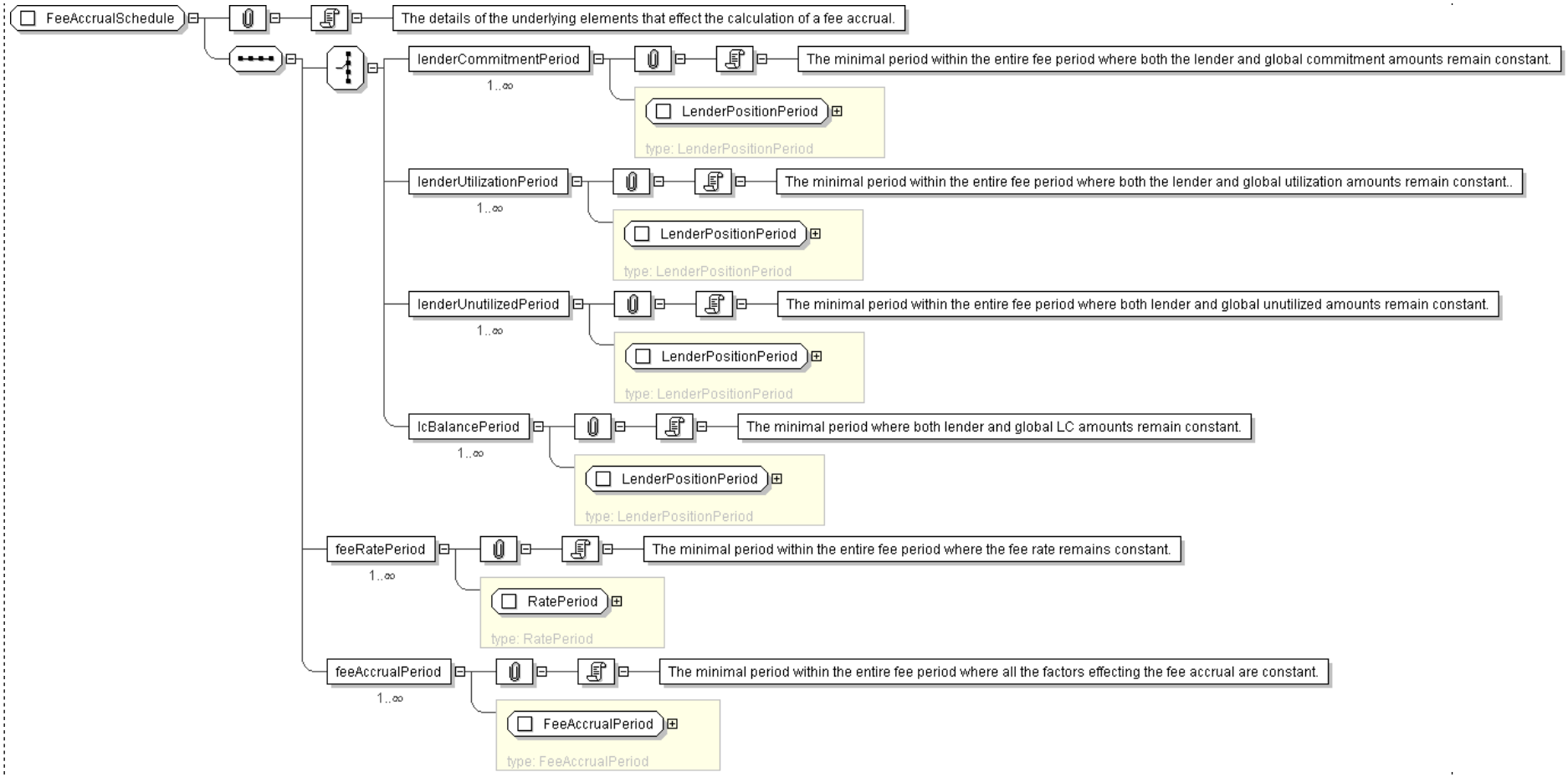
Super-types:	None
Sub-types:	None

Name	FeeAccrualSchedule
Used by (from the same schema document)	Complex Type OnGoingFeeNotice
Abstract	no
Documentation	The details of the underlying elements that effect the calculation of a fee accrual.

XML Instance Representation

```
<...>
  Start Choice [1]
    <lenderCommitmentPeriod> LenderPositionPeriod </lenderCommitmentPeriod> [1..*]
    'The minimal period within the entire fee period where both the lender and global commitment amounts remain constant.'

    <lenderUtilizationPeriod> LenderPositionPeriod </lenderUtilizationPeriod> [1..*]
    'The minimal period within the entire fee period where both the lender and global utilization amounts remain constant..'

    <lenderUnutilizedPeriod> LenderPositionPeriod </lenderUnutilizedPeriod> [1..*]
    'The minimal period within the entire fee period where both lender and global unutilized amounts remain constant.'

    <lcBalancePeriod> LenderPositionPeriod </lcBalancePeriod> [1..*]
    'The minimal period where both lender and global LC amounts remain constant.'

  End Choice
  <feeRatePeriod> RatePeriod </feeRatePeriod> [1..*]
  'The minimal period within the entire fee period where the fee rate remains constant.'

  <feeAccrualPeriod> FeeAccrualPeriod </feeAccrualPeriod> [1..*]
  'The minimal period within the entire fee period where all the factors effecting the fee accrual are constant.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FeeAccrualSchedule">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="lenderCommitmentPeriod" type=" LenderPositionPeriod " maxOccurs="unbounded"/>
      <xsd:element name="lenderUtilizationPeriod" type=" LenderPositionPeriod " maxOccurs="unbounded"/>
      <xsd:element name="lenderUnutilizedPeriod" type=" LenderPositionPeriod " maxOccurs="unbounded"/>
      <xsd:element name="lcBalancePeriod" type=" LenderPositionPeriod " maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:element name="feeRatePeriod" type=" RatePeriod " maxOccurs="unbounded"/>
    <xsd:element name="feeAccrualPeriod" type=" FeeAccrualPeriod " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxTerms

[Table of contents]

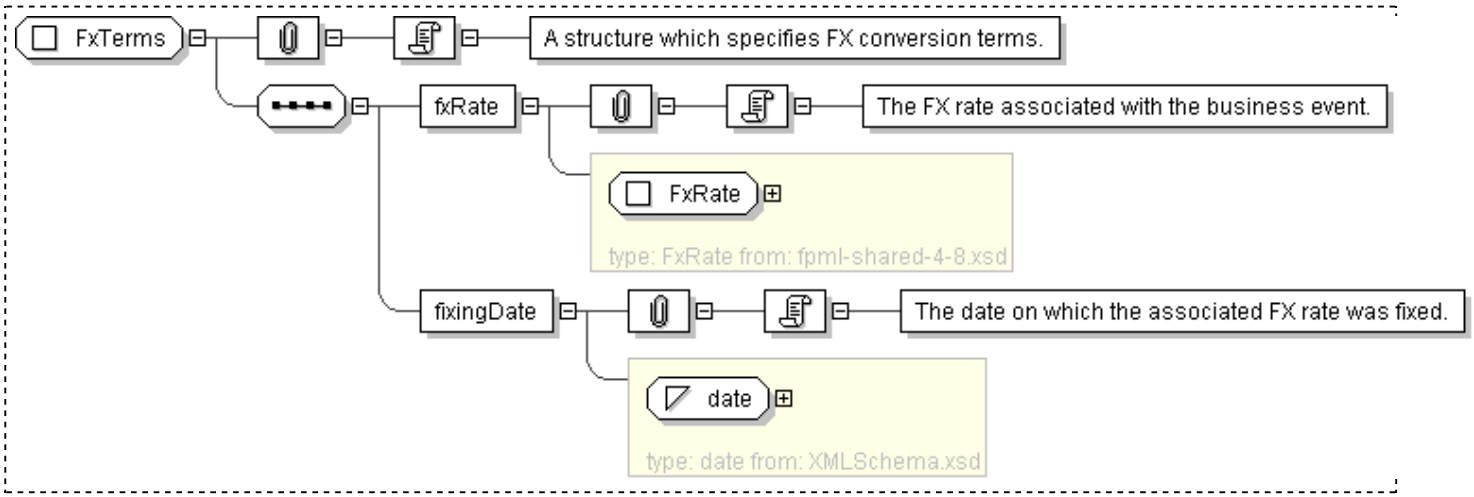
Super-types:	None
Sub-types:	None

Name	FxTerms
Used by (from the same schema document)	Complex Type FxTermsSchedule , Complex Type LetterOfCredit
Abstract	no
Documentation	A structure which specifies FX conversion terms.

XML Instance Representation

```
<...>
   <fxRate> FxRate </fxRate> [1]
   'The FX rate associated with the business event.'

   <fixingDate> xsd:date </fixingDate> [1]
   'The date on which the associated FX rate was fixed.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxTerms">
   <xsd:sequence>
      <xsd:element name="fxRate" type=" FxRate "/>
      <xsd:element name="fixingDate" type=" xsd:date "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxTermsSchedule

[Table of contents]

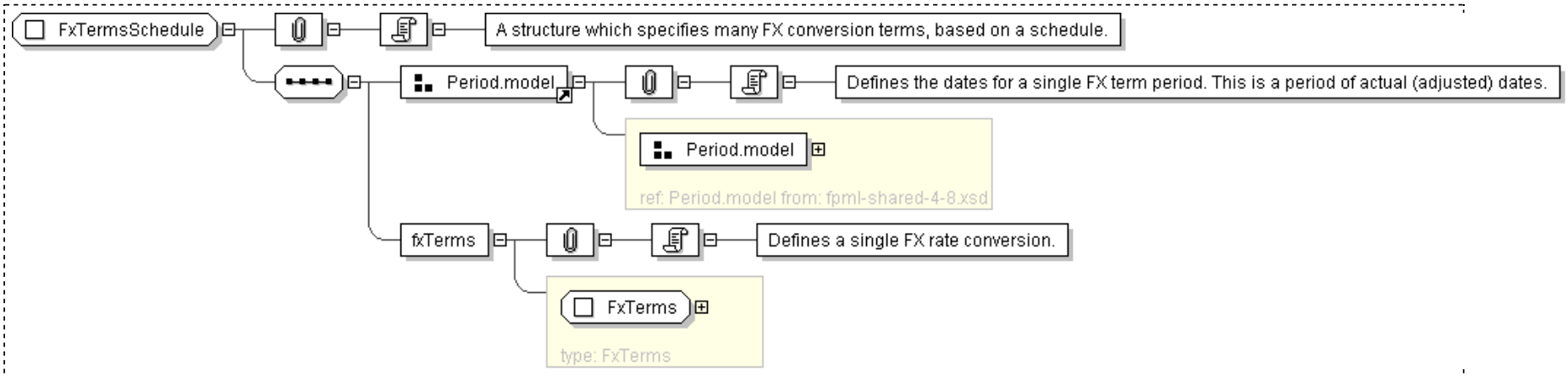
Super-types:	None
Sub-types:	None

Name	FxTermsSchedule
Used by (from the same schema document)	Complex Type LoanContract
Abstract	no
Documentation	A structure which specifies many FX conversion terms, based on a schedule.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
    'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
    'Date on which this period ends.'

   <fxTerms> FxTerms </fxTerms> [1]
    'Defines a single FX rate conversion.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxTermsSchedule">
   <xsd:sequence>
      <xsd:group ref=" Period.model "/>
```

```
    <xsd:element name="fxTerms" type=" FxTerms "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestAccrualPeriod

[Table of contents]

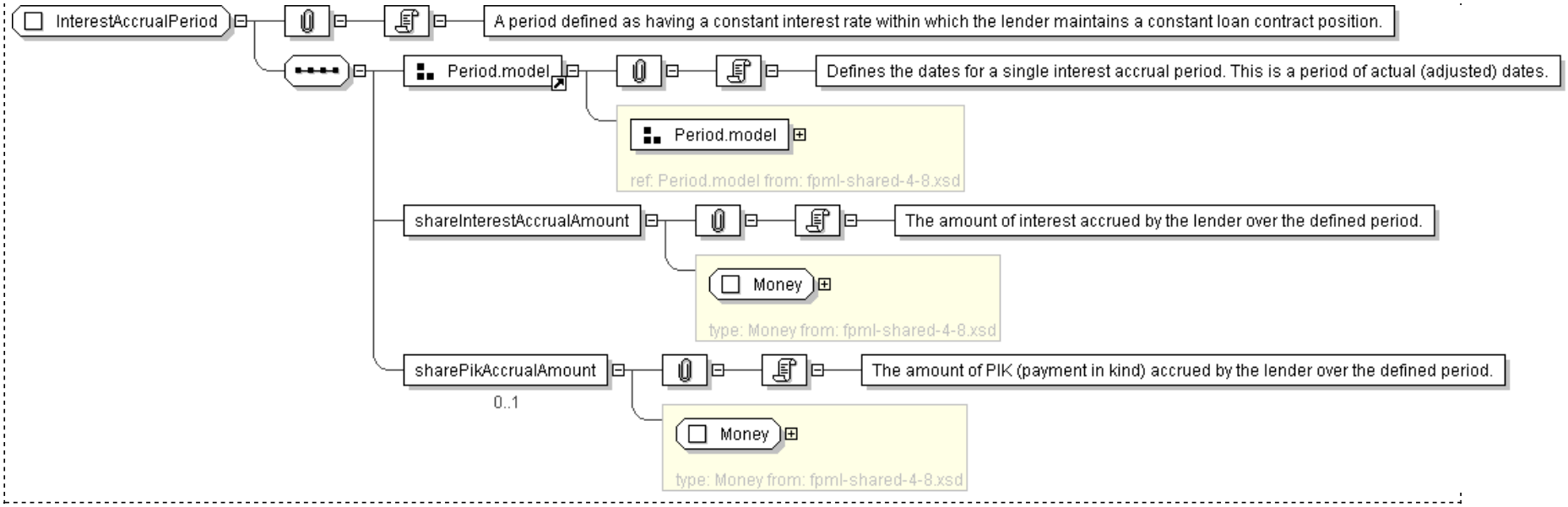
Super-types:	None
Sub-types:	None

Name	InterestAccrualPeriod
Used by (from the same schema document)	Complex Type InterestAccrualSchedule
Abstract	no
Documentation	A period defined as having a constant interest rate within which the lender maintains a constant loan contract position.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
    'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
    'Date on which this period ends.'

   <shareInterestAccrualAmount> Money </shareInterestAccrualAmount> [1]
    'The amount of interest accrued by the lender over the defined period.'

   <sharePikAccrualAmount> Money </sharePikAccrualAmount> [0..1]
    'The amount of PIK (payment in kind) accrued by the lender over the defined period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestAccrualPeriod">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:element name="shareInterestAccrualAmount" type=" Money "/>
    <xsd:element name="sharePikAccrualAmount" type=" Money " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestAccrualSchedule

[Table of contents]

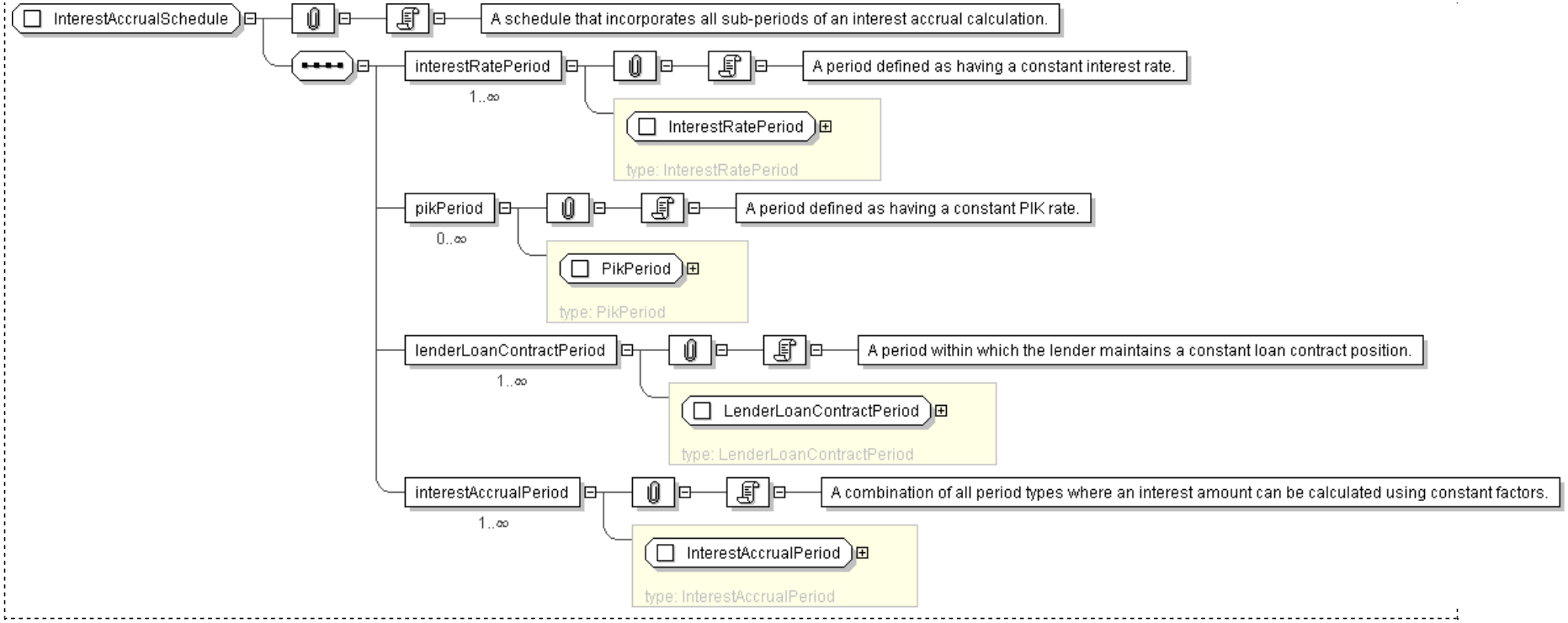
Super-types:	None
Sub-types:	None

Name	InterestAccrualSchedule
Used by (from the same schema document)	Model Group InterestPaymentDetails.model
Abstract	no
Documentation	A schedule that incorporates all sub-periods of an interest accrual calculation.

XML Instance Representation

```
<...>
   <interestRatePeriod> InterestRatePeriod </interestRatePeriod> [1..*]
   'A period defined as having a constant interest rate.'

   <pikPeriod> PikPeriod </pikPeriod> [0..*]
   'A period defined as having a constant PIK rate.'

   <lenderLoanContractPeriod> LenderLoanContractPeriod </lenderLoanContractPeriod> [1..*]
   'A period within which the lender maintains a constant loan contract position.'

   <interestAccrualPeriod> InterestAccrualPeriod </interestAccrualPeriod> [1..*]
   'A combination of all period types where an interest amount can be calculated using constant factors.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestAccrualSchedule">
  <xsd:sequence>
    <xsd:element name="interestRatePeriod" type=" InterestRatePeriod " maxOccurs="unbounded"/>
    <xsd:element name="pikPeriod" type=" PikPeriod " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="lenderLoanContractPeriod" type=" LenderLoanContractPeriod " maxOccurs="unbounded"/>
    <xsd:element name="interestAccrualPeriod" type=" InterestAccrualPeriod " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestPayment

[Table of contents]

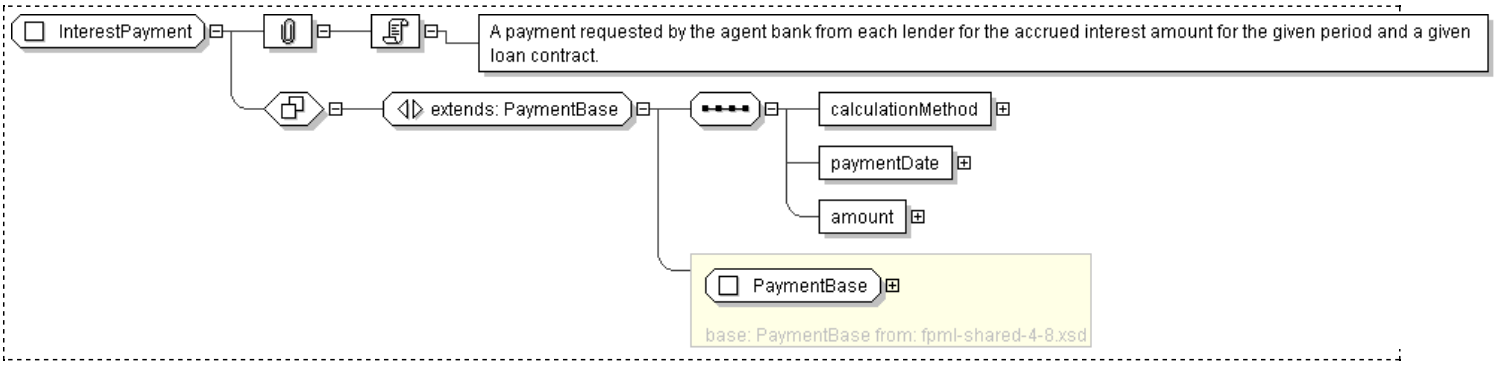
Super-types:	PaymentBase < InterestPayment (by extension)
Sub-types:	None

Name	InterestPayment
Used by (from the same schema document)	Model Group InterestPaymentDetails.model
Abstract	no
Documentation	A payment requested by the agent bank from each lender for the accrued interest amount for the given period and a given loan contract.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <calculationMethod> InterestCalculationMethodEnum </calculationMethod> [1]
   'Defines the way in which the agent bank is allocating interest payments - can be (i) pro-rata at the time of
   the interest payment or (ii) based on the loan contract share throughout the interest period (which is the
   preferred method).'

   <paymentDate> xsd:date </paymentDate> [1]
   'The date on which interest was paid to the lender by the agent bank. This is an actual (adjusted) date.'

   <amount> ParticipationAmount </amount> [1]
   'The amount of interest paid by the borrower to the agent bank and share paid to the lender.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestPayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="calculationMethod" type=" InterestCalculationMethodEnum "/>
        <xsd:element name="paymentDate" type=" xsd:date "/>
        <xsd:element name="amount" type=" ParticipationAmount "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestPaymentNotice

[Table of contents]

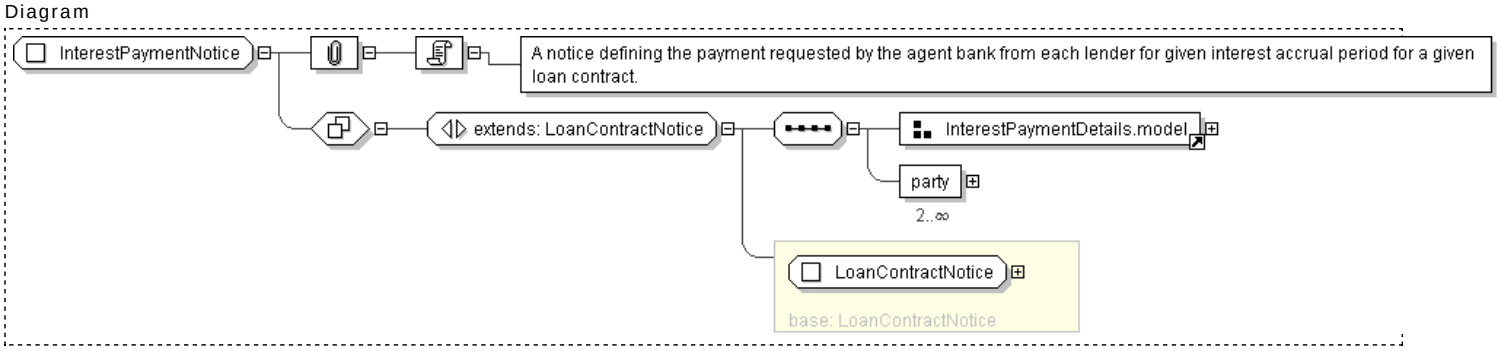
Super-types:	NotificationMessage < LoanContractNotice (by extension) < InterestPaymentNotice (by extension)
Sub-types:	None

Name	InterestPaymentNotice
Abstract	no
Documentation	A notice defining the payment requested by the agent bank from each lender for given interest accrual period for a given loan contract.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead,
 it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build
 number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a
 version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases
 within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan contract levels.
   This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional information.'

   Start Choice [1]
 'The sender may choose to either transmit the full or partial loan contract details.'


      <loanContract> LoanContract </loanContract> [1]
      'A core structure describing a loan contract between borrower and lenders forming part or all of the credit
      line offered by a facility structure within a deal.'

      <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
      'A basic set of fields used to uniquely identify the loan contract.'

   End Choice
   <interestPayment> InterestPayment </interestPayment> [1]
   'Represents the total amount of interest paid by the borrower to the agent bank and the share of it paid to the
   lender.'

   <interestAccrualSchedule> InterestAccrualSchedule </interestAccrualSchedule> [1]
   'A schedule that incorporates all sub-periods of an interest accrual calculation.'

   <party> Party </party> [2..*]
   'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestPaymentNotice">
  <xsd:complexContent>
    <xsd:extension base=" LoanContractNotice ">
      <xsd:sequence>
        <xsd:group ref=" InterestPaymentDetails.model "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestRatePeriod

[Table of contents]

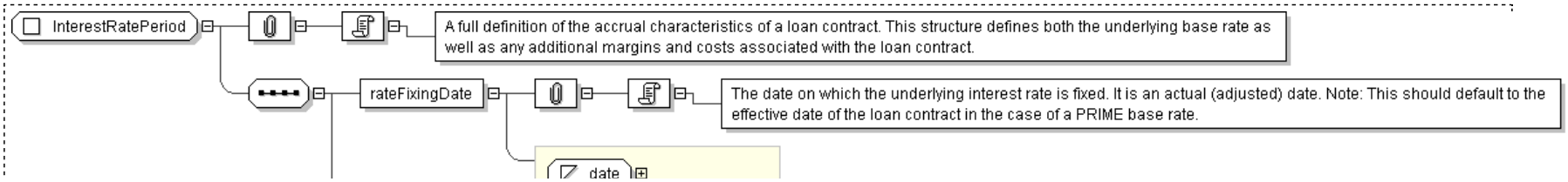
Super-types:	None
Sub-types:	None

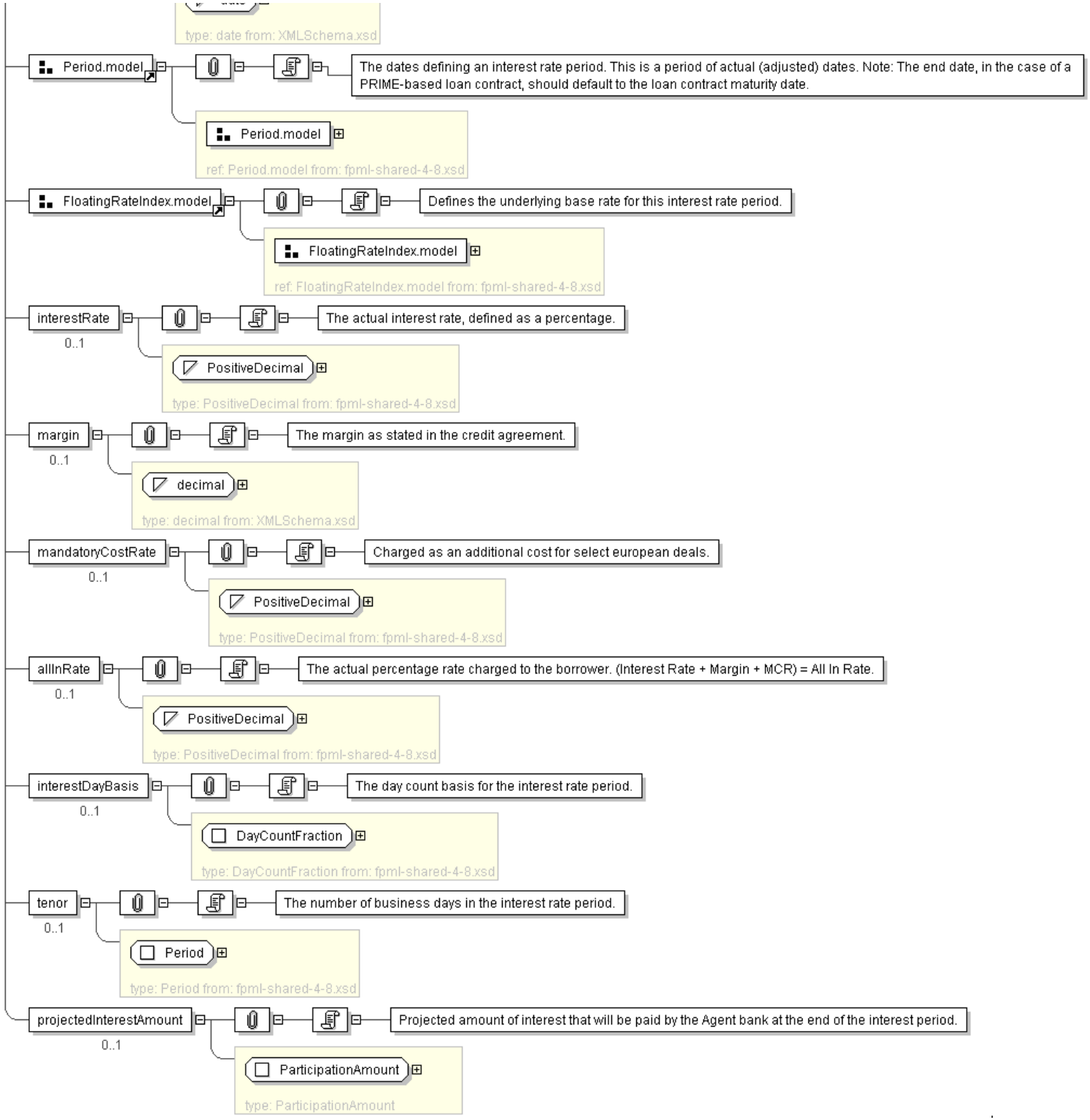
Name	InterestRatePeriod
Used by (from the same schema document)	Complex Type InterestAccrualSchedule , Complex Type LoanContract
Abstract	no
Documentation	A full definition of the accrual characteristics of a loan contract. This structure defines both the underlying base rate as well as any additional margins and costs associated with the loan contract.

XML Instance Representation

```
<...>
   <rateFixingDate> xsd:date </rateFixingDate> [1]
   'The date on which the underlying interest rate is fixed. It is an actual (adjusted) date. Note: This should default to the effective date
   of the loan contract in the case of a PRIME base rate.'

   <startDate> xsd:date </startDate> [1]
   'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
   'Date on which this period ends.'

   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <interestRate> PositiveDecimal </interestRate> [0..1]
   'The actual interest rate, defined as a percentage.'

   <margin> xsd:decimal </margin> [0..1]
   'The margin as stated in the credit agreement.'

   <mandatoryCostRate> PositiveDecimal </mandatoryCostRate> [0..1]
   'Charged as an additional cost for select european deals.'

   <allInRate> PositiveDecimal </allInRate> [0..1]
   'The actual percentage rate charged to the borrower. (Interest Rate + Margin + MCR) = All In Rate.'

   <interestDayBasis> DayCountFraction </interestDayBasis> [0..1]
   'The day count basis for the interest rate period.'

   <tenor> Period </tenor> [0..1]
   'The number of business days in the interest rate period.'

   <projectedInterestAmount> ParticipationAmount </projectedInterestAmount> [0..1]
   'Projected amount of interest that will be paid by the Agent bank at the end of the interest period.'

</...>
```

Diagram




type: date from: XMLSchema.xsd
Period.model
The dates defining an interest rate period. This is a period of actual (adjusted) dates. Note: The end date, in the case of a PRIME-based loan contract, should default to the loan contract maturity date.
Period.model
ref: Period.model from: fpml-shared-4-8.xsd
FloatingRateIndex.model
Defines the underlying base rate for this interest rate period.
FloatingRateIndex.model
ref: FloatingRateIndex.model from: fpml-shared-4-8.xsd
interestRate
0..1
The actual interest rate, defined as a percentage.
PositiveDecimal
type: PositiveDecimal from: fpml-shared-4-8.xsd
margin
0..1
The margin as stated in the credit agreement.
decimal
type: decimal from: XMLSchema.xsd
mandatoryCostRate
0..1
Charged as an additional cost for select european deals.
PositiveDecimal
type: PositiveDecimal from: fpml-shared-4-8.xsd
allInRate
0..1
The actual percentage rate charged to the borrower. (Interest Rate + Margin + MCR) = All In Rate.
PositiveDecimal
type: PositiveDecimal from: fpml-shared-4-8.xsd
interestDayBasis
0..1
The day count basis for the interest rate period.
DayCountFraction
type: DayCountFraction from: fpml-shared-4-8.xsd
tenor
0..1
The number of business days in the interest rate period.
Period
type: Period from: fpml-shared-4-8.xsd
projectedInterestAmount
0..1
Projected amount of interest that will be paid by the Agent bank at the end of the interest period.
ParticipationAmount
type: ParticipationAmount

Schema Component Representation

```
<xsd:complexType name="InterestRatePeriod">
   <xsd:sequence>
      <xsd:element name="rateFixingDate" type=" xsd:date "/>
      <xsd:group ref=" Period.model "/>
      <xsd:group ref=" FloatingRateIndex.model "/>
      <xsd:element name="interestRate" type=" PositiveDecimal " minOccurs="0"/>
      <xsd:element name="margin" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="mandatoryCostRate" type=" PositiveDecimal " minOccurs="0"/>
      <xsd:element name="allInRate" type=" PositiveDecimal " minOccurs="0"/>
      <xsd:element name="interestDayBasis" type=" DayCountFraction " minOccurs="0"/>
      <xsd:element name="tenor" type=" Period " minOccurs="0"/>
      <xsd:element name="projectedInterestAmount" type=" ParticipationAmount " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcAmendmentNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < LcAmendmentNotice (by extension)
Sub-types:	None

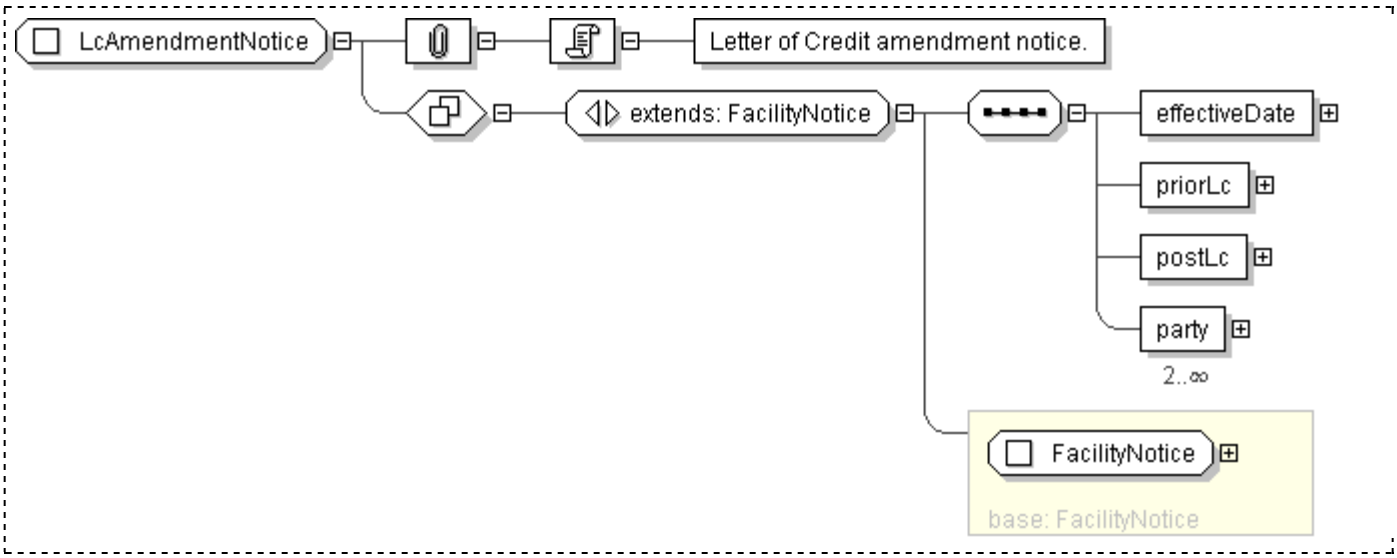
Name	LcAmendmentNotice
Abstract	no
Documentation	Letter of Credit amendment notice.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'
```

```
<facilityCommitmentPosition> FacilityCommitmentPosition
</facilityCommitmentPosition> [0..1]
'A structure which contains the position being held by the lender on both the
facility and loan contract levels. This position information is from the
message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
'A flag which can be set by the message sender in order to signify an
exceptional business event.'

<comments> xsd:string </comments> [0..1]
'A free-form, manually entered field which will be used by users directly for
additional information.'

<effectiveDate> xsd:date </effectiveDate> [1]
'Effective date of the amendment.'

<priorLc> LetterOfCredit </priorLc> [1]
'The letter of credit before the amendment.'

<postLc> LetterOfCredit </postLc> [1]
'The letter of credit after the amendment.'

<party> Party </party> [2..*]
'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcAmendmentNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:element name="effectiveDate" type=" xsd:date "/>
        <xsd:element name="priorLc" type=" LetterOfCredit "/>
        <xsd:element name="postLc" type=" LetterOfCredit "/>
        <xsd:element name="party" type=" Party " minOccurs="2"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcBalanceNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < LcBalanceNotice (by extension)
Sub-types:	None

Name	LcBalanceNotice
Abstract	no
Documentation	Letter of Credit balance change notice.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'
```

<facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]

'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]

'A flag which can be set by the message sender in order to signify an exceptional business event.'

<comments> xsd:string </comments> [0..1]

'A free-form, manually entered field which will be used by users directly for additional information.'

<lcSummary> LcSummary </lcSummary> [1]

'A basic set of fields to identify an LC. Please note that the partyReference within the contract identifier should refer to the Issuing Bank.'

<effectiveDate> xsd:date </effectiveDate> [1]

'Effective date of the balance change.'

<priorAmount> ParticipationAmount </priorAmount> [0..1]

'The global and lender share amount of the L/C before any associated balance change.'

<balanceChangeAmount> ParticipationAmount </balanceChangeAmount> [1]

'The amount that the L/C has shifted by. Shown at both the global and share levels.'

<currentAmount> ParticipationAmount </currentAmount> [1]

'The global and lender share amount of the L/C after any associated balance change.'

<lcDrawFlag> xsd:boolean </lcDrawFlag> [1]

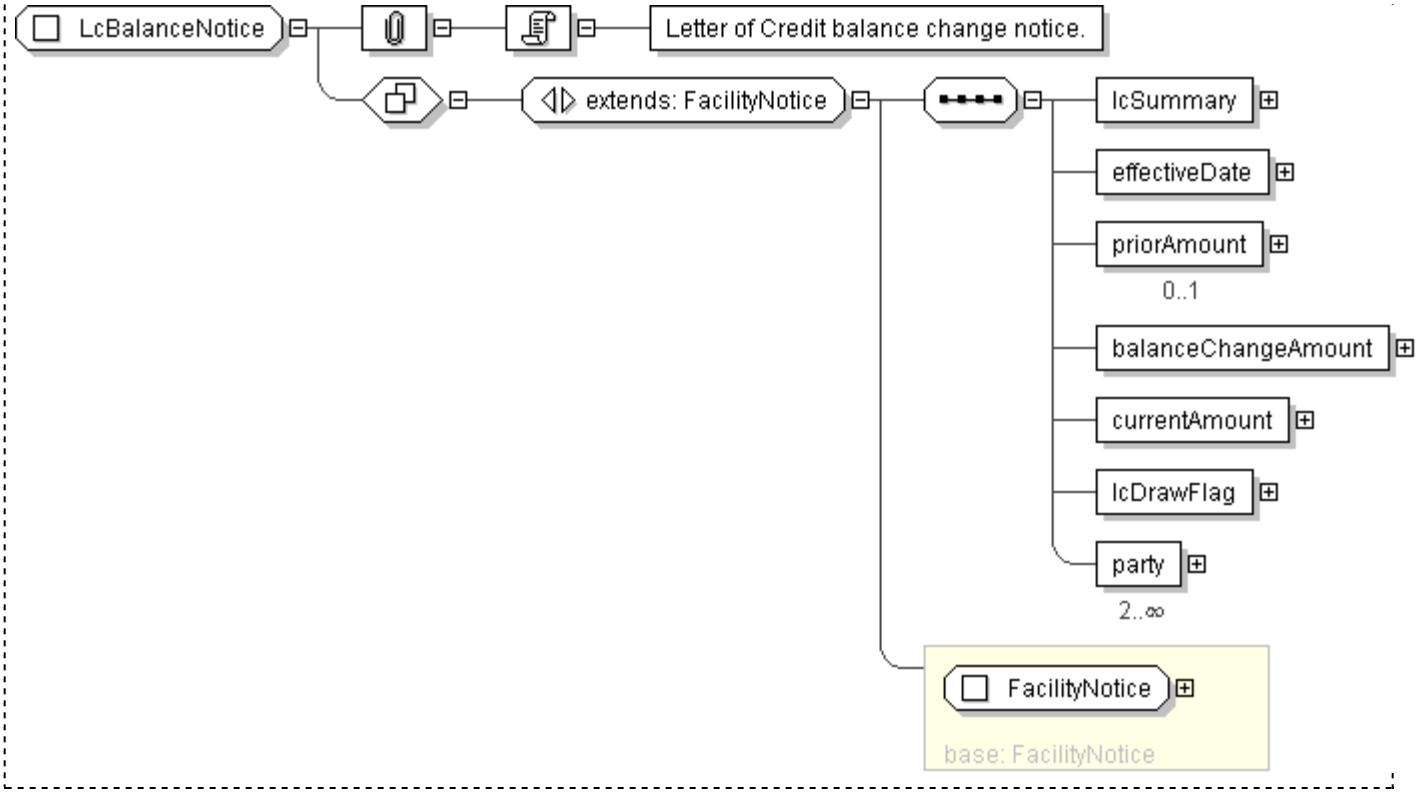
'Set to false for a normal increase or decrease in the LC amount. An LC draw results in a reduction of the LC amount and signifies that a loan drawdown will take place in conjunction with this message.'

<party> Party </party> [2..*]

'The parties involved with the associated transaction.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="LcBalanceNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:element name="lcSummary" type=" LcSummary "/>
        <xsd:element name="effectiveDate" type=" xsd:date "/>
        <xsd:element name="priorAmount" type=" ParticipationAmount " minOccurs="0"/>
        <xsd:element name="balanceChangeAmount" type=" ParticipationAmount "/>
        <xsd:element name="currentAmount" type=" ParticipationAmount "/>
        <xsd:element name="lcDrawFlag" type=" xsd:boolean "/>
        <xsd:element name="party" type=" Party " minOccurs="2"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcEvergreenOption

[Table of contents]

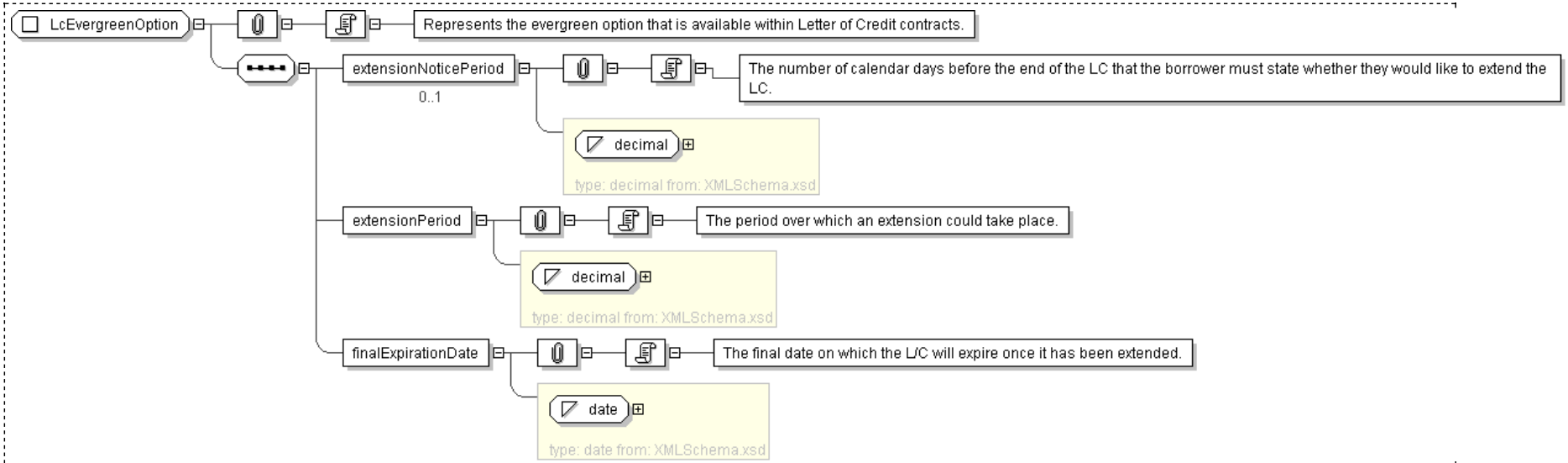
Super-types:	None
Sub-types:	None

Name	LcEvergreenOption
Used by (from the same schema document)	Complex Type LetterOfCredit
Abstract	no
Documentation	Represents the evergreen option that is available within Letter of Credit contracts.

XML Instance Representation

```
<...>
  <extensionNoticePeriod> xsd:decimal </extensionNoticePeriod> [0..1]
  'The number of calendar days before the end of the LC that the borrower must state whether they would like to extend the LC.'

  <extensionPeriod> xsd:decimal </extensionPeriod> [1]
  'The period over which an extension could take place.'

  <finalExpirationDate> xsd:date </finalExpirationDate> [1]
  'The final date on which the L/C will expire once it has been extended.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcEvergreenOption">
  <xsd:sequence>
    <xsd:element name="extensionNoticePeriod" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="extensionPeriod" type=" xsd:decimal "/>
    <xsd:element name="finalExpirationDate" type=" xsd:date "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcIssuanceNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < LcIssuanceNotice (by extension)
Sub-types:	None

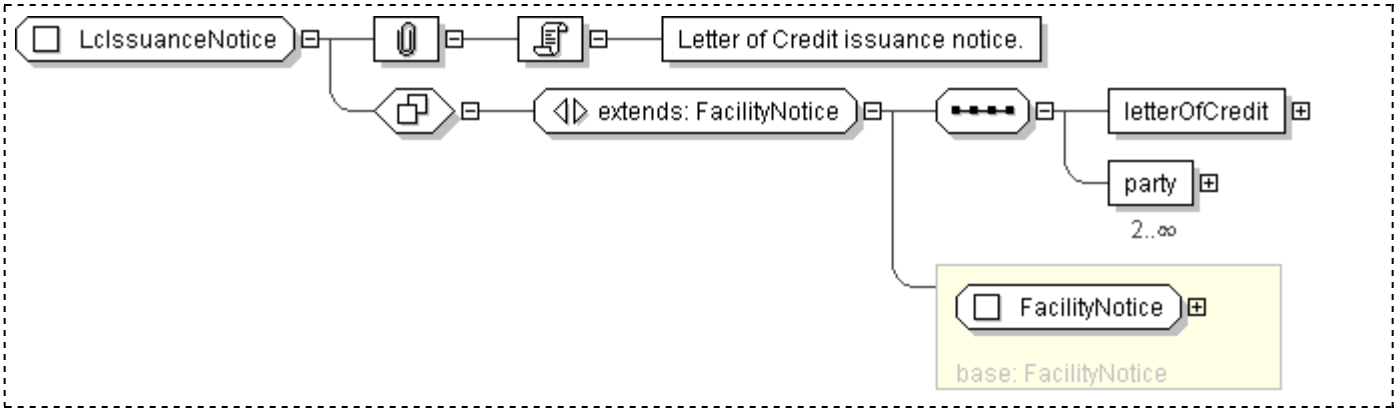
Name	LcIssuanceNotice
Abstract	no
Documentation	Letter of Credit issuance notice.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'
```

```
<facilityCommitmentPosition> FacilityCommitmentPosition
</facilityCommitmentPosition> [0..1]
'A structure which contains the position being held by the lender on both the
facility and loan contract levels. This position information is from the
message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
'A flag which can be set by the message sender in order to signify an
exceptional business event.'

<comments> xsd:string </comments> [0..1]
'A free-form, manually entered field which will be used by users directly for
additional information.'

<letterOfCredit> LetterOfCredit </letterOfCredit> [1]
<party> Party </party> [2..*]
'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcIssuanceNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:element name="letterOfCredit" type=" LetterOfCredit "/>
        <xsd:element name="party" type=" Party " minOccurs="2"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcPosition

[Table of contents]

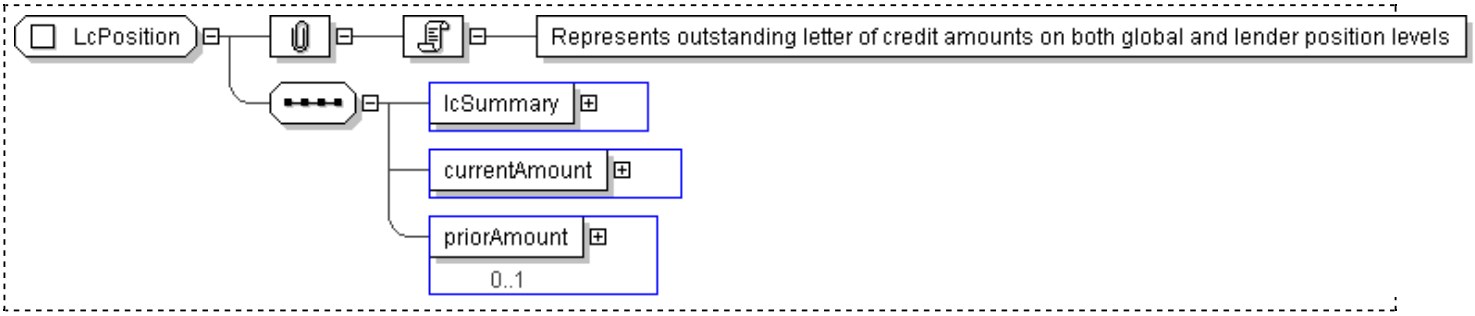
Super-types:	None
Sub-types:	None

Name	LcPosition
Used by (from the same schema document)	Complex Type FacilityCommitmentPosition
Abstract	no
Documentation	Represents outstanding letter of credit amounts on both global and lender position levels

XML Instance Representation

```
<...>
   <lcSummary> LcSummary </lcSummary> [1]
   <currentAmount> ParticipationAmount </currentAmount> [1]
   <priorAmount> ParticipationAmount </priorAmount> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcPosition">
   <xsd:sequence>
      <xsd:element name="lcSummary" type=" LcSummary "/>
      <xsd:element name="currentAmount" type=" ParticipationAmount "/>
      <xsd:element name="priorAmount" type=" ParticipationAmount " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcSummary

[Table of contents]

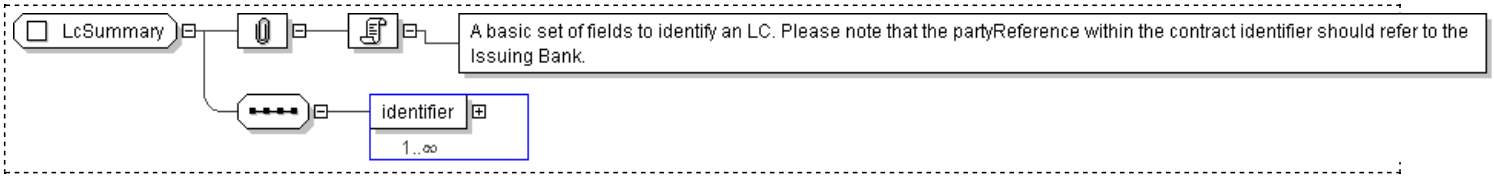
Super-types:	None
Sub-types:	None

Name	LcSummary
Used by (from the same schema document)	Complex Type LcBalanceNotice , Complex Type LcPosition , Complex Type LcTerminationNotice , Complex Type LetterOfCredit , Complex Type OneOffFeeNotice , Complex Type OnGoingFeeRateChange , Complex Type OnGoingFeeNotice
Abstract	no
Documentation	A basic set of fields to identify an LC. Please note that the partyReference within the contract identifier should refer to the Issuing Bank.

XML Instance Representation

```
<...>
  <identifier> ContractIdentifier </identifier> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcSummary">
  <xsd:sequence>
    <xsd:element name="identifier" type=" ContractIdentifier " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LcTerminationNotice

[Table of contents]

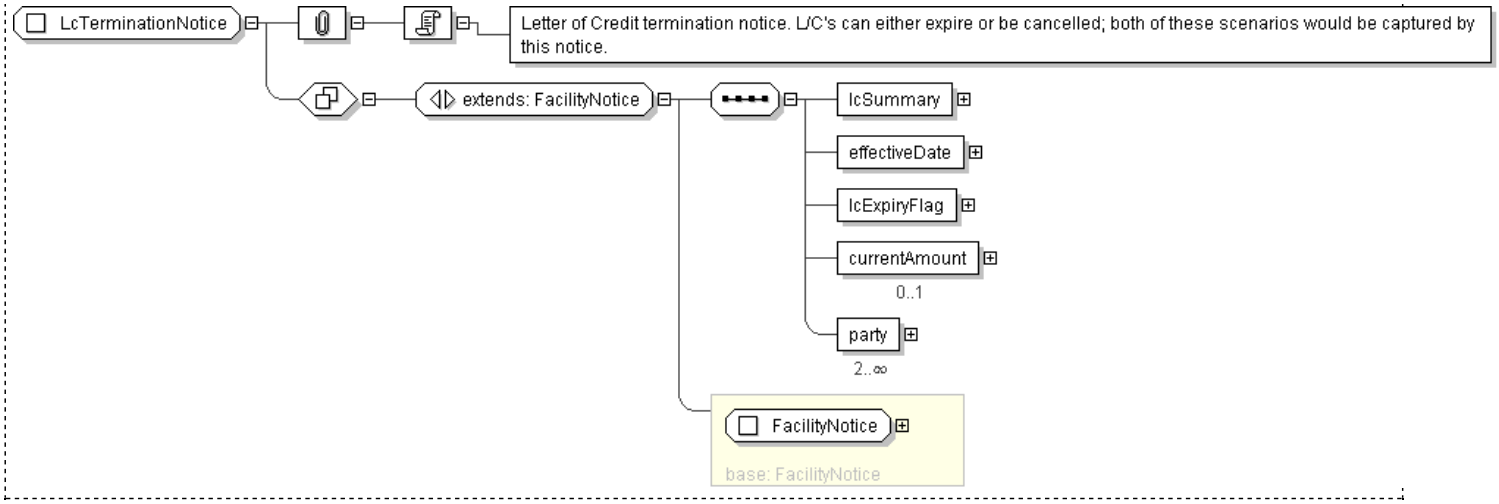
Super-types:	NotificationMessage < FacilityNotice (by extension) < LcTerminationNotice (by extension)
Sub-types:	None

Name	LcTerminationNotice
Abstract	no
Documentation	Letter of Credit termination notice. L/C's can either expire or be cancelled; both of these scenarios would be captured by this notice.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the
 build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples
 within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between
 releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
 same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan contract levels.
   This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional information.'

   <lcSummary> LcSummary </lcSummary> [1]
   <effectiveDate> xsd:date </effectiveDate> [1]
   <lcExpiryFlag> xsd:boolean </lcExpiryFlag> [1]
   'Determines whether this event was created due to a natural expiration of the LC or an unscheduled
   cancellation.'

   <currentAmount> ParticipationAmount </currentAmount> [0..1]
   <party> Party </party> [2..*]
   'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LcTerminationNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:element name="lcSummary" type=" LcSummary "/>
        <xsd:element name="effectiveDate" type=" xsd:date "/>
        <xsd:element name="lcExpiryFlag" type=" xsd:boolean "/>
        <xsd:element name="currentAmount" type=" ParticipationAmount " minOccurs="0"/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LenderLoanContractPeriod

[Table of contents]

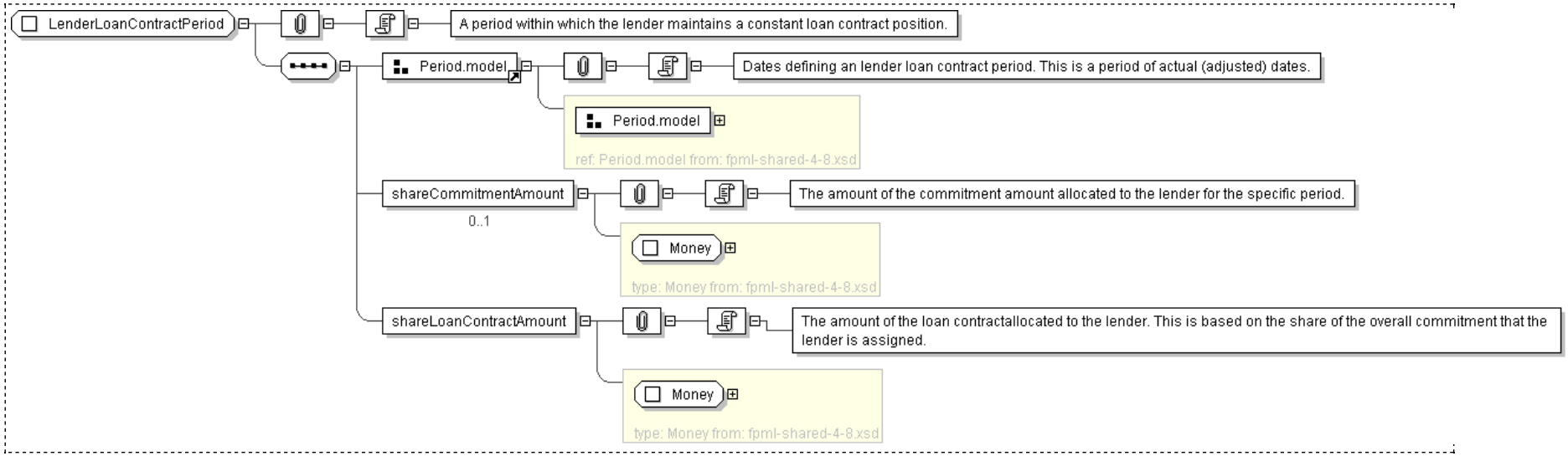
Super-types:	None
Sub-types:	None

Name	LenderLoanContractPeriod
Used by (from the same schema document)	Complex Type InterestAccrualSchedule
Abstract	no
Documentation	A period within which the lender maintains a constant loan contract position.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
   'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
   'Date on which this period ends.'

   <shareCommitmentAmount> Money </shareCommitmentAmount> [0..1]
   'The amount of the commitment amount allocated to the lender for the specific period.'

   <shareLoanContractAmount> Money </shareLoanContractAmount> [1]
   'The amount of the loan contractallocated to the lender. This is based on the share of the overall commitment that the lender is assigned.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LenderLoanContractPeriod">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:element name="shareCommitmentAmount" type=" Money " minOccurs="0"/>
    <xsd:element name="shareLoanContractAmount" type=" Money "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LenderPositionPeriod

[Table of contents]

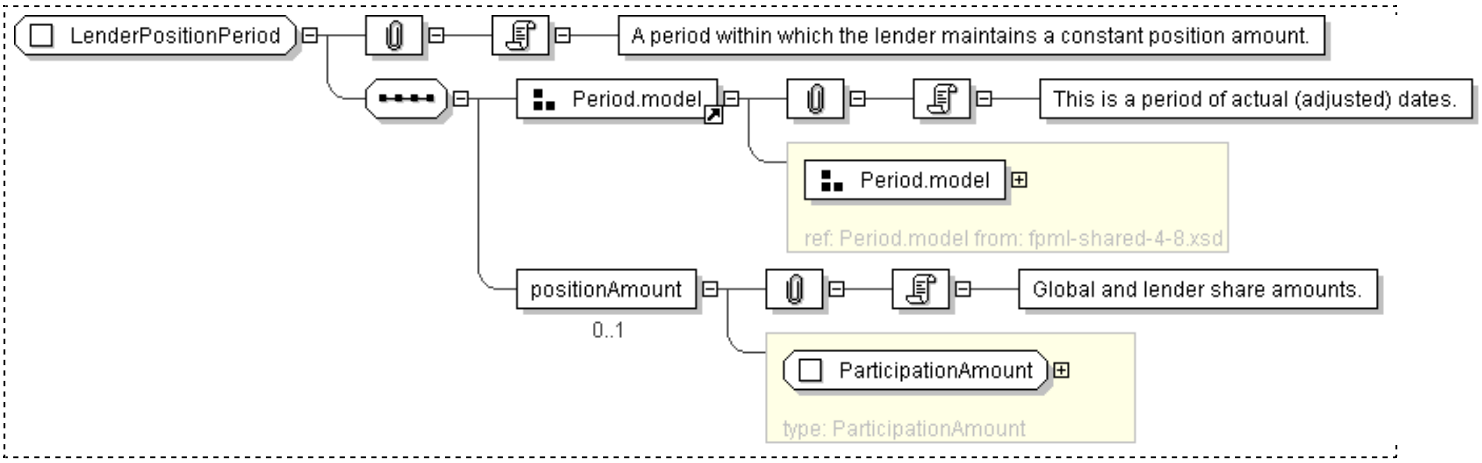
Super-types:	None
Sub-types:	None

Name	LenderPositionPeriod
Used by (from the same schema document)	Complex Type FeeAccrualSchedule , Complex Type FeeAccrualSchedule , Complex Type FeeAccrualSchedule , Complex Type FeeAccrualSchedule
Abstract	no
Documentation	A period within which the lender maintains a constant position amount.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
   'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
   'Date on which this period ends.'

   <positionAmount> ParticipationAmount </positionAmount> [0..1]
   'Global and lender share amounts.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LenderPositionPeriod">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:element name="positionAmount" type=" ParticipationAmount " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LetterOfCredit

[Table of contents]

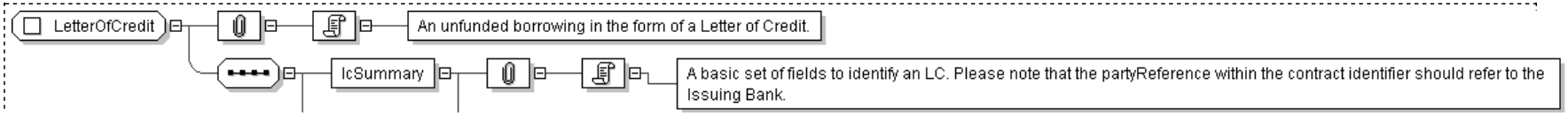
Super-types:	None
Sub-types:	None

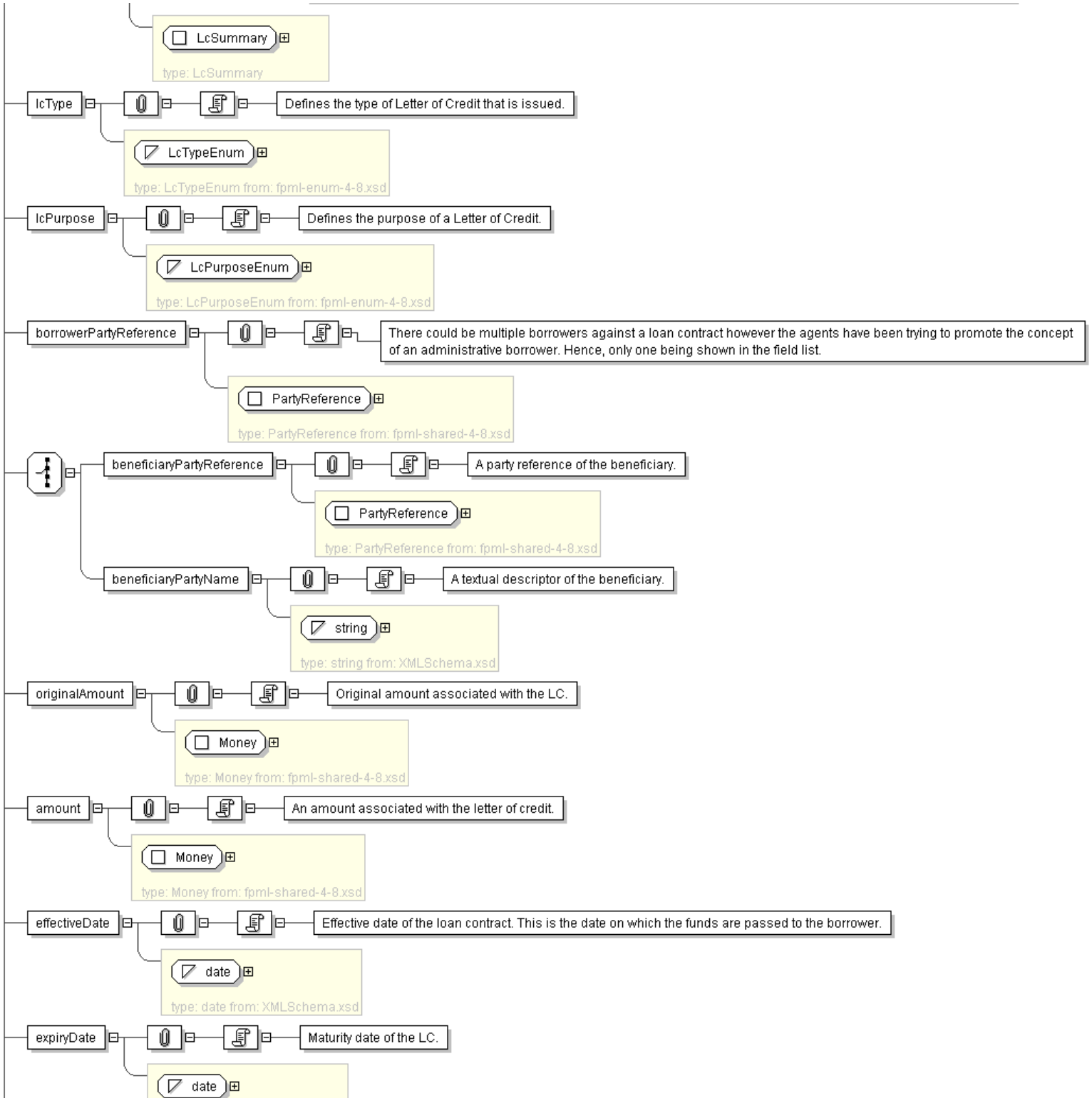
Name	LetterOfCredit
Used by (from the same schema document)	Complex Type LcAmendmentNotice , Complex Type LcAmendmentNotice , Complex Type LcIssuanceNotice
Abstract	no
Documentation	An unfunded borrowing in the form of a Letter of Credit.

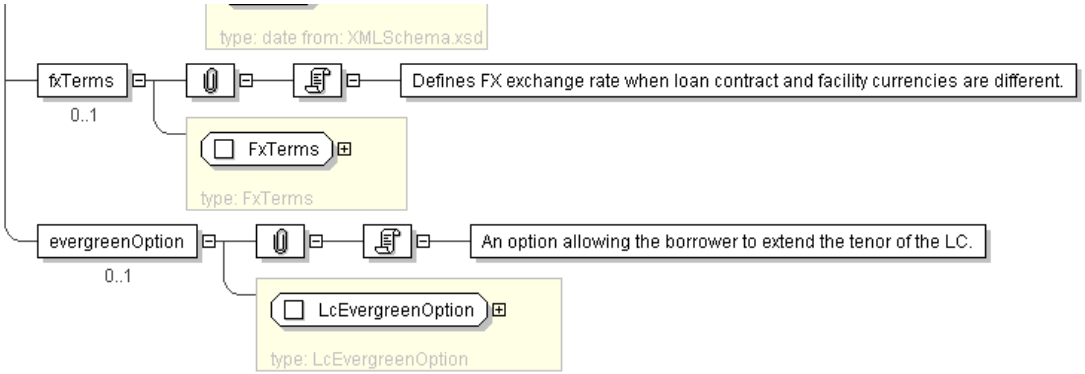
XML Instance Representation

```
<...>
   <lcSummary> LcSummary </lcSummary> [1]
   'A basic set of fields to identify an LC. Please note that the partyReference within the contract identifier should refer to the Issuing
   Bank.'

   <lcType> LcTypeEnum </lcType> [1]
   'Defines the type of Letter of Credit that is issued.'

   <lcPurpose> LcPurposeEnum </lcPurpose> [1]
   'Defines the purpose of a Letter of Credit.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [1]
   'There could be multiple borrowers against a loan contract however the agents have been trying to promote the concept of an administrative
   borrower. Hence, only one being shown in the field list.'

   Start Choice [1]
      <beneficiaryPartyReference> PartyReference </beneficiaryPartyReference> [1]
      'A party reference of the beneficiary.'

      <beneficiaryPartyName> xsd:string </beneficiaryPartyName> [1]
      'A textual descriptor of the beneficiary.'

   End Choice
   <originalAmount> Money </originalAmount> [1]
   'Original amount associated with the LC.'

   <amount> Money </amount> [1]
   'An amount associated with the letter of credit.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'Effective date of the loan contract. This is the date on which the funds are passed to the borrower.'

   <expiryDate> xsd:date </expiryDate> [1]
   'Maturity date of the LC.'

   <fxTerms> FxTerms </fxTerms> [0..1]
   'Defines FX exchange rate when loan contract and facility currencies are different.'

   <evergreenOption> LcEvergreenOption </evergreenOption> [0..1]
   'An option allowing the borrower to extend the tenor of the LC.'

</...>
```

Diagram




LcSummary
type: LcSummary
lcType
Defines the type of Letter of Credit that is issued.
LcTypeEnum
type: LcTypeEnum from: fpml-enum-4-8.xsd
lcPurpose
Defines the purpose of a Letter of Credit.
LcPurposeEnum
type: LcPurposeEnum from: fpml-enum-4-8.xsd
borrowerPartyReference
There could be multiple borrowers against a loan contract however the agents have been trying to promote the concept of an administrative borrower. Hence, only one being shown in the field list.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
beneficiaryPartyReference
A party reference of the beneficiary.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
beneficiaryPartyName
A textual descriptor of the beneficiary.
string
type: string from: XMLSchema.xsd
originalAmount
Original amount associated with the LC.
Money
type: Money from: fpml-shared-4-8.xsd
amount
An amount associated with the letter of credit.
Money
type: Money from: fpml-shared-4-8.xsd
effectiveDate
Effective date of the loan contract. This is the date on which the funds are passed to the borrower.
date
type: date from: XMLSchema.xsd
expiryDate
Maturity date of the LC.
date



Schema Component Representation

```
<xsd:complexType name="LetterOfCredit">
  <xsd:sequence>
    <xsd:element name="lcSummary" type=" LcSummary "/>
    <xsd:element name="lcType" type=" LcTypeEnum "/>
    <xsd:element name="lcPurpose" type=" LcPurposeEnum "/>
    <xsd:element name="borrowerPartyReference" type=" PartyReference "/>
    <xsd:choice>
      <xsd:element name="beneficiaryPartyReference" type=" PartyReference "/>
      <xsd:element name="beneficiaryPartyName" type=" xsd:string "/>
    </xsd:choice>
    <xsd:element name="originalAmount" type=" Money "/>
    <xsd:element name="amount" type=" Money "/>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
    <xsd:element name="expiryDate" type=" xsd:date "/>
    <xsd:element name="fxTerms" type=" FxTerms " minOccurs="0"/>
    <xsd:element name="evergreenOption" type=" LcEvergreenOption " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanContract

[Table of contents]

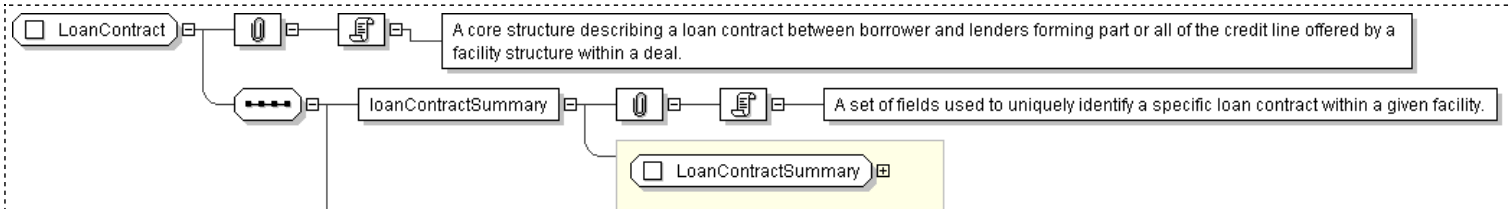
Super-types:	None
Sub-types:	None

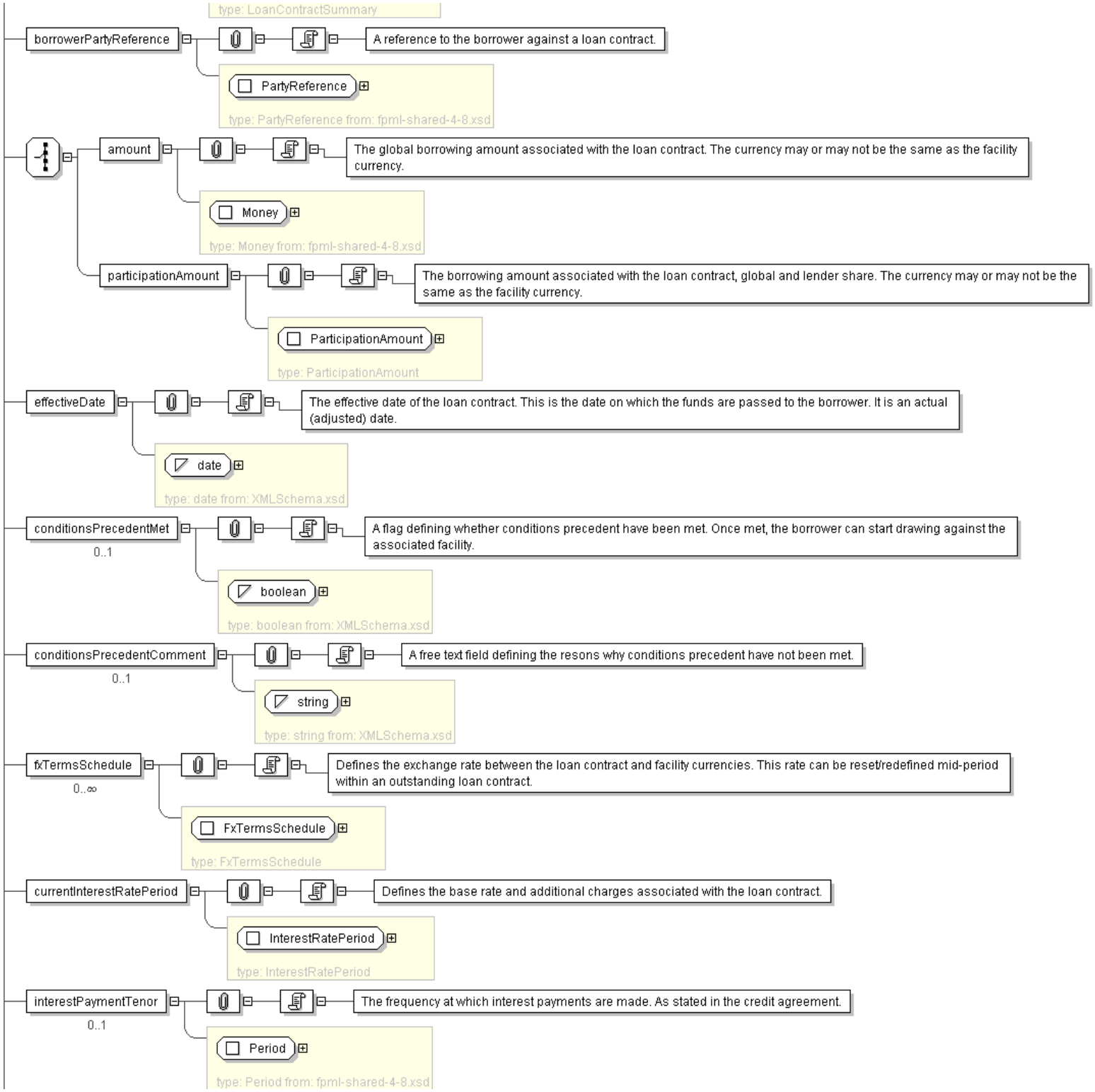
Name	LoanContract
Used by (from the same schema document)	Complex Type LoanContractNotice , Complex Type NewLoanContracts
Abstract	no
Documentation	A core structure describing a loan contract between borrower and lenders forming part or all of the credit line offered by a facility structure within a deal.

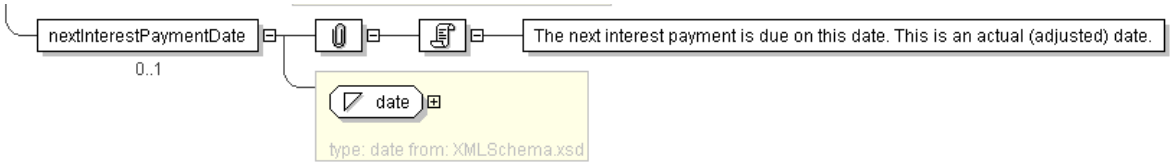
XML Instance Representation

```
<...>
  <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
   'A set of fields used to uniquely identify a specific loan contract within a given facility.'

  <borrowerPartyReference> PartyReference </borrowerPartyReference> [1]
   'A reference to the borrower against a loan contract.'

  Start Choice [1]
    <amount> Money </amount> [1]
     'The global borrowing amount associated with the loan contract. The currency may or may not be the same as the facility currency.'

    <participationAmount> ParticipationAmount </participationAmount> [1]
     'The borrowing amount associated with the loan contract, global and lender share. The currency may or may not be the same as the facility
     currency.'

  End Choice
  <effectiveDate> xsd:date </effectiveDate> [1]
   'The effective date of the loan contract. This is the date on which the funds are passed to the borrower. It is an actual (adjusted) date.'

  <conditionsPrecedentMet> xsd:boolean </conditionsPrecedentMet> [0..1]
   'A flag defining whether conditions precedent have been met. Once met, the borrower can start drawing against the associated facility.'

  <conditionsPrecedentComment> xsd:string </conditionsPrecedentComment> [0..1]
   'A free text field defining the resons why conditions precedent have not been met.'

  <fxTermsSchedule> FxTermsSchedule </fxTermsSchedule> [0..*]
   'Defines the exchange rate between the loan contract and facility currencies. This rate can be reset/redefined mid-period within an outstanding
   loan contract.'

  <currentInterestRatePeriod> InterestRatePeriod </currentInterestRatePeriod> [1]
   'Defines the base rate and additional charges associated with the loan contract.'

  <interestPaymentTenor> Period </interestPaymentTenor> [0..1]
   'The frequency at which interest payments are made. As stated in the credit agreement.'

  <nextInterestPaymentDate> xsd:date </nextInterestPaymentDate> [0..1]
   'The next interest payment is due on this date. This is an actual (adjusted) date.'

</...>
```

Diagram




type: LoanContractSummary
borrowerPartyReference
A reference to the borrower against a loan contract.
PartyReference
type: PartyReference from: fpml-shared-4-8.xsd
amount
The global borrowing amount associated with the loan contract. The currency may or may not be the same as the facility currency.
Money
type: Money from: fpml-shared-4-8.xsd
participationAmount
The borrowing amount associated with the loan contract, global and lender share. The currency may or may not be the same as the facility currency.
ParticipationAmount
type: ParticipationAmount
effectiveDate
The effective date of the loan contract. This is the date on which the funds are passed to the borrower. It is an actual (adjusted) date.
date
type: date from: XMLSchema.xsd
conditionsPrecedentMet
0..1
A flag defining whether conditions precedent have been met. Once met, the borrower can start drawing against the associated facility.
boolean
type: boolean from: XMLSchema.xsd
conditionsPrecedentComment
0..1
A free text field defining the resons why conditions precedent have not been met.
string
type: string from: XMLSchema.xsd
fxTermsSchedule
0..∞
Defines the exchange rate between the loan contract and facility currencies. This rate can be reset/redefined mid-period within an outstanding loan contract.
FxTermsSchedule
type: FxTermsSchedule
currentInterestRatePeriod
Defines the base rate and additional charges associated with the loan contract.
InterestRatePeriod
type: InterestRatePeriod
interestPaymentTenor
0..1
The frequency at which interest payments are made. As stated in the credit agreement.
Period
type: Period from: fpml-shared-4-8.xsd



Schema Component Representation

```
<xsd:complexType name="LoanContract">
  <xsd:sequence>
    <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
    <xsd:element name="borrowerPartyReference" type=" PartyReference "/>
    <xsd:choice>
      <xsd:element name="amount" type=" Money "/>
      <xsd:element name="participationAmount" type=" ParticipationAmount "/>
    </xsd:choice>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
    <xsd:element name="conditionsPrecedentMet" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="conditionsPrecedentComment" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="fxTermsSchedule" type=" FxTermsSchedule " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="currentInterestRatePeriod" type=" InterestRatePeriod "/>
    <xsd:element name="interestPaymentTenor" type=" Period " minOccurs="0"/>
    <xsd:element name="nextInterestPaymentDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanContractNotice

[Table of contents]

Super-types:	NotificationMessage < LoanContractNotice (by extension)
Sub-types:	• DrawdownNotice (by extension) • InterestPaymentNotice (by extension)

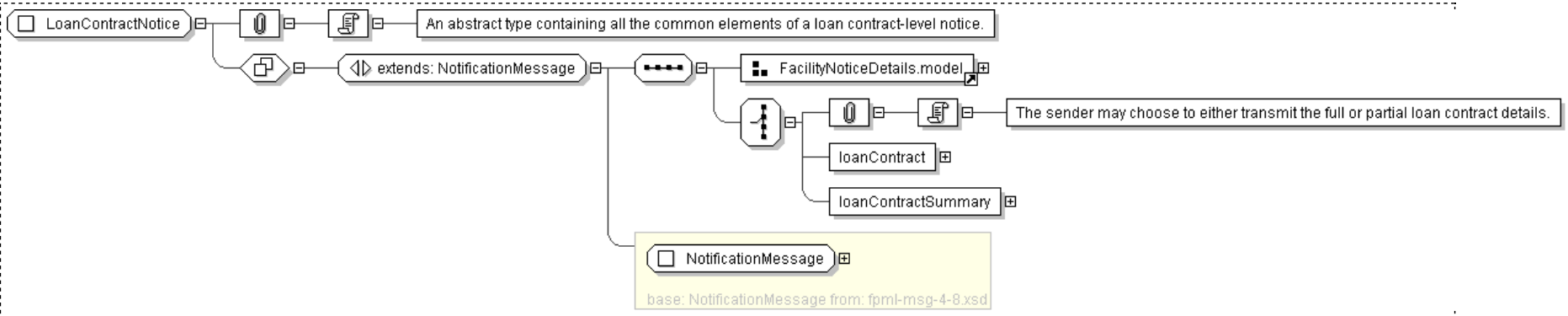
Name	LoanContractNotice
Abstract	yes
Documentation	An abstract type containing all the common elements of a loan contract-level notice.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to
 define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML
 parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a
 change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is
   from the message sender\'s viewpoint as of the date of the associated notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional information.'
```

```
  Start Choice [1]
'The sender may choose to either transmit the full or partial loan contract details.'

    <loanContract> LoanContract </loanContract> [1]
    'A core structure describing a loan contract between borrower and lenders forming part or all of the credit line offered by a facility
    structure within a deal.'

    <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
    'A basic set of fields used to uniquely identify the loan contract.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LoanContractNotice" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:group ref=" FacilityNoticeDetails.model "/>
        <xsd:choice>
          <xsd:element name="loanContract" type=" LoanContract "/>
          <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanContractPosition

[Table of contents]

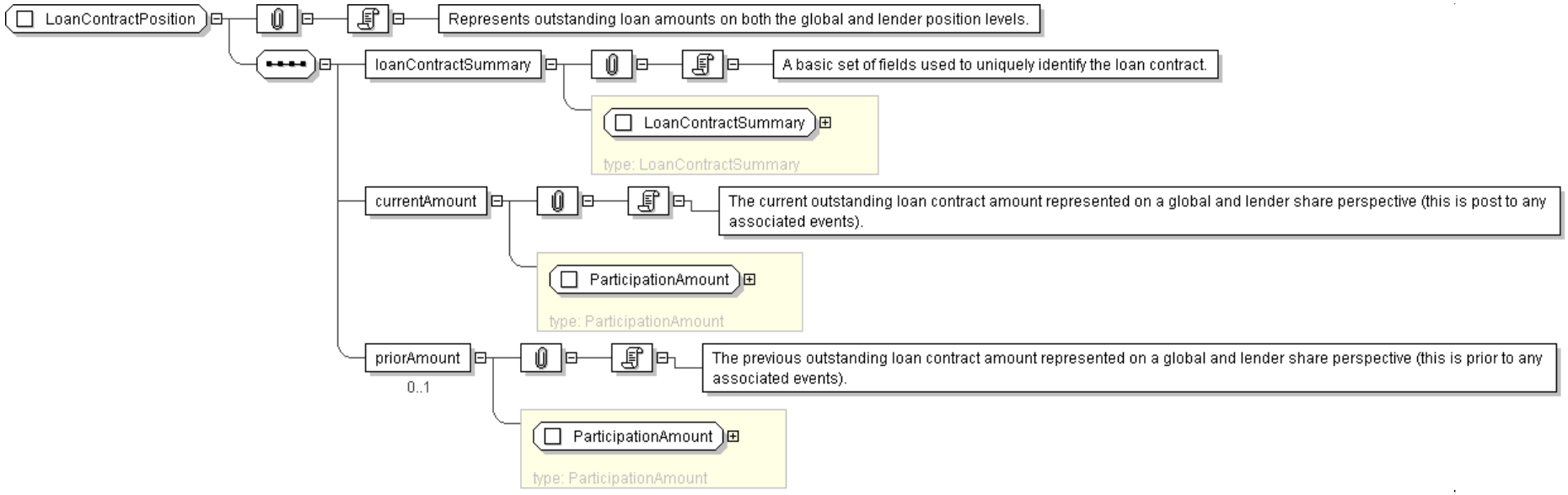
Super-types:	None
Sub-types:	None

Name	LoanContractPosition
Used by (from the same schema document)	Complex Type FacilityCommitmentPosition
Abstract	no
Documentation	Represents outstanding loan amounts on both the global and lender position levels.

XML Instance Representation

```
<...>
  <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
   'A basic set of fields used to uniquely identify the loan contract.'

  <currentAmount> ParticipationAmount </currentAmount> [1]
   'The current outstanding loan contract amount represented on a global and lender share perspective (this is post to any associated
   events).'

  <priorAmount> ParticipationAmount </priorAmount> [0..1]
   'The previous outstanding loan contract amount represented on a global and lender share perspective (this is prior to any associated
   events).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LoanContractPosition">
  <xsd:sequence>
```

```
      <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
      <xsd:element name="currentAmount" type=" ParticipationAmount "/>
      <xsd:element name="priorAmount" type=" ParticipationAmount " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanContractRepayment

[Table of contents]

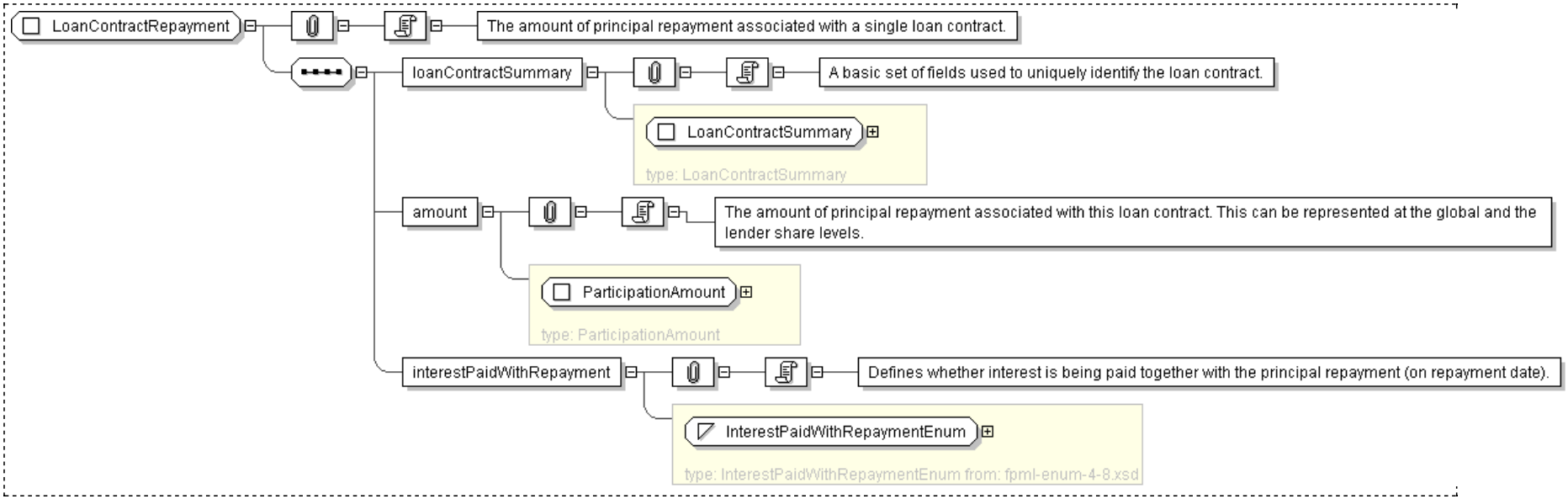
Super-types:	None
Sub-types:	None

Name	LoanContractRepayment
Used by (from the same schema document)	Complex Type Repayment
Abstract	no
Documentation	The amount of principal repayment associated with a single loan contract.

XML Instance Representation

```
<...>
  <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
  'A basic set of fields used to uniquely identify the loan contract.'

  <amount> ParticipationAmount </amount> [1]
  'The amount of principal repayment associated with this loan contract. This can be represented at the global and the lender share
  levels.'

  <interestPaidWithRepayment> InterestPaidWithRepaymentEnum </interestPaidWithRepayment> [1]
  'Defines whether interest is being paid together with the principal repayment (on repayment date).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LoanContractRepayment">
  <xsd:sequence>
```

```
    <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
    <xsd:element name="amount" type=" ParticipationAmount "/>
    <xsd:element name="interestPaidWithRepayment" type=" InterestPaidWithRepaymentEnum "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LoanContractSummary

[Table of contents]

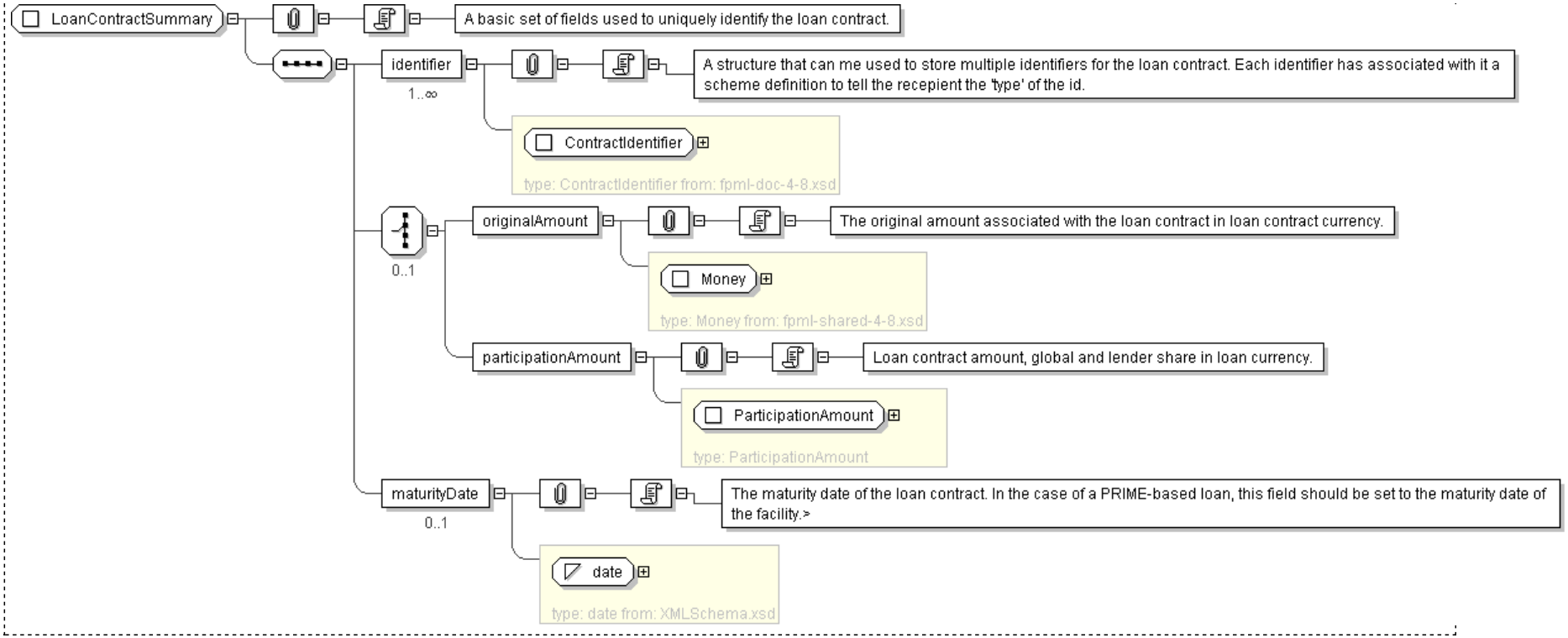
Super-types:	None
Sub-types:	None

Name	LoanContractSummary
Used by (from the same schema document)	Complex Type LoanContract , Complex Type LoanContractNotice , Complex Type LoanContractPosition , Complex Type LoanContractRepayment , Complex Type MarginRateChange , Complex Type MaturingLoanContract , Complex Type OneOffFeeNotice , Complex Type OnGoingFeeNotice
Abstract	no
Documentation	A basic set of fields used to uniquely identify the loan contract.

XML Instance Representation

```
<...>
   <identifier> ContractIdentifier </identifier> [1..*]
   'A structure that can me used to store multiple identifiers for the loan contract. Each identifier has associated with it a scheme
   definition to tell the recepient the \'type\' of the id.'

   Start Choice [0..1]
      <originalAmount> Money </originalAmount> [1]
      'The original amount associated with the loan contract in loan contract currency.'

      <participationAmount> ParticipationAmount </participationAmount> [1]
      'Loan contract amount, global and lender share in loan currency.'

   End Choice
   <maturityDate> xsd:date </maturityDate> [0..1]
   'The maturity date of the loan contract. In the case of a PRIME-based loan, this field should be set to the maturity date of the
   facility.>'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LoanContractSummary">
  <xsd:sequence>
    <xsd:element name="identifier" type=" ContractIdentifier " maxOccurs="unbounded"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="originalAmount" type=" Money "/>
      <xsd:element name="participationAmount" type=" ParticipationAmount "/>
    </xsd:choice>
    <xsd:element name="maturityDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MarginRateChange

[Table of contents]

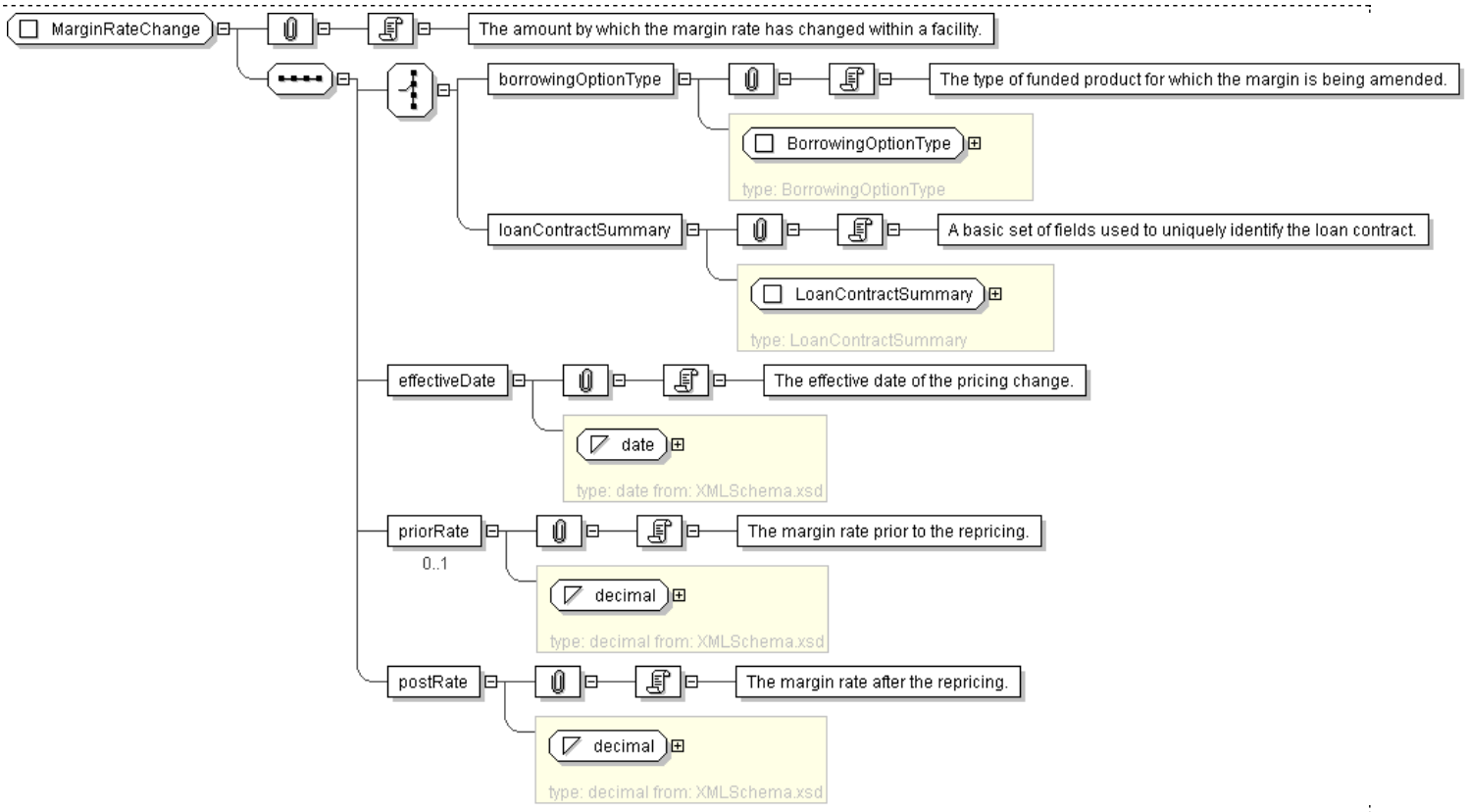
Super-types:	None
Sub-types:	None

Name	MarginRateChange
Used by (from the same schema document)	Complex Type PricingChangeNotice
Abstract	no
Documentation	The amount by which the margin rate has changed within a facility.

XML Instance Representation

```
<...>
  Start Choice [1]
    <borrowingOptionType> BorrowingOptionType </borrowingOptionType> [1]
    'The type of funded product for which the margin is being amended.'

    <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
    'A basic set of fields used to uniquely identify the loan contract.'

  End Choice
  <effectiveDate> xsd:date </effectiveDate> [1]
  'The effective date of the pricing change.'

  <priorRate> xsd:decimal </priorRate> [0..1]
  'The margin rate prior to the repricing.'

  <postRate> xsd:decimal </postRate> [1]
  'The margin rate after the repricing.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MarginRateChange">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="borrowingOptionType" type=" BorrowingOptionType "/>
      <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
    </xsd:choice>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
```

```
      <xsd:element name="priorRate" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="postRate" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MaturingLoanContract

[Table of contents]

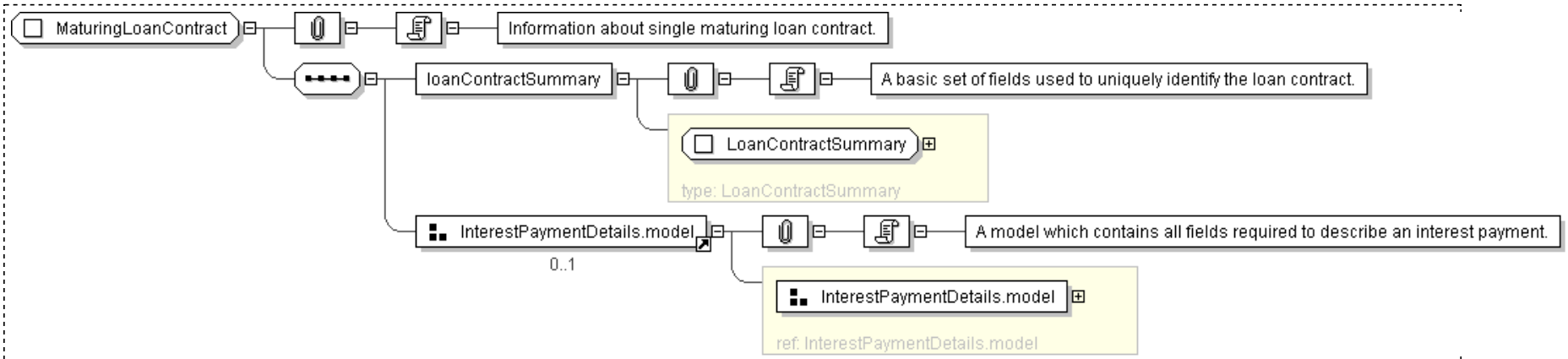
Super-types:	None
Sub-types:	None

Name	MaturingLoanContract
Used by (from the same schema document)	Complex Type MaturingLoanContracts
Abstract	no
Documentation	Information about single maturing loan contract.

XML Instance Representation

```
<...>
  <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
   'A basic set of fields used to uniquely identify the loan contract.'

  Start Group: InterestPaymentDetails.model [0..1]
'A model which contains all fields required to describe an interest payment.'


     <interestPayment> InterestPayment </interestPayment> [1]
     'Represents the total amount of interest paid by the borrower to the agent bank and the share of it paid to the
     lender.'

     <interestAccrualSchedule> InterestAccrualSchedule </interestAccrualSchedule> [1]
     'A schedule that incorporates all sub-periods of an interest accrual calculation.'

  End Group: InterestPaymentDetails.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MaturingLoanContract">
  <xsd:sequence>
    <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
    <xsd:group ref=" InterestPaymentDetails.model " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MaturingLoanContracts

[Table of contents]

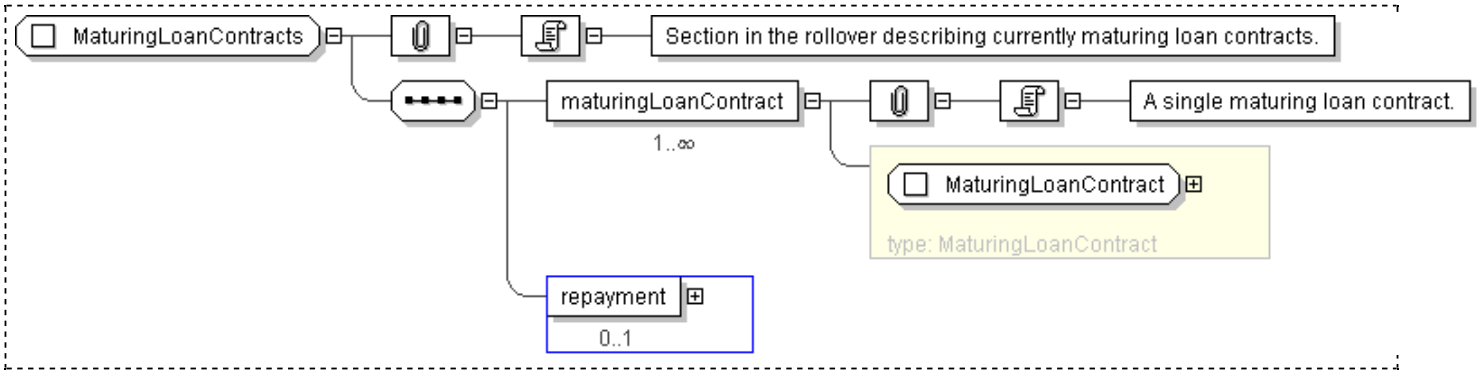
Super-types:	None
Sub-types:	None

Name	MaturingLoanContracts
Used by (from the same schema document)	Complex Type RolloverNotice
Abstract	no
Documentation	Section in the rollover describing currently maturing loan contracts.

XML Instance Representation

```
<...>
   <maturingLoanContract> MaturingLoanContract </maturingLoanContract> [1..*]
   'A single maturing loan contract.'

   <repayment> Repayment </repayment> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MaturingLoanContracts">
   <xsd:sequence>
      <xsd:element name="maturingLoanContract" type=" MaturingLoanContract "
      maxOccurs="unbounded"/>
      <xsd:element name="repayment" type=" Repayment " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NewLoanContracts

[Table of contents]

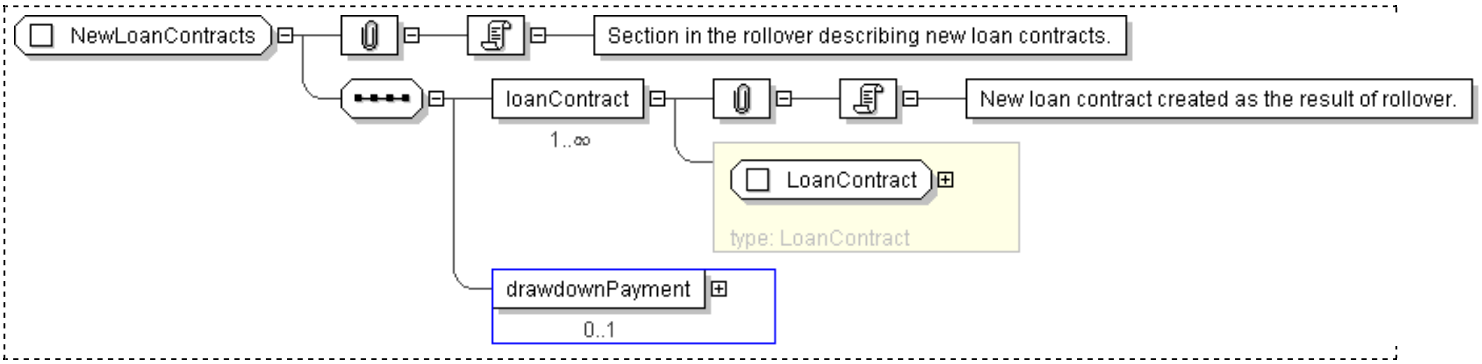
Super-types:	None
Sub-types:	None

Name	NewLoanContracts
Used by (from the same schema document)	Complex Type RolloverNotice
Abstract	no
Documentation	Section in the rollover describing new loan contracts.

XML Instance Representation

```
<...>
   <loanContract> LoanContract </loanContract> [1..*]
   'New loan contract created as the result of rollover.'

   <drawdownPayment> DrawdownPayment </drawdownPayment> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NewLoanContracts">
   <xsd:sequence>
      <xsd:element name="loanContract" type=" LoanContract " maxOccurs="unbounded"/>
      <xsd:element name="drawdownPayment" type=" DrawdownPayment " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OneOffFeeNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < OneOffFeeNotice (by extension)
Sub-types:	None

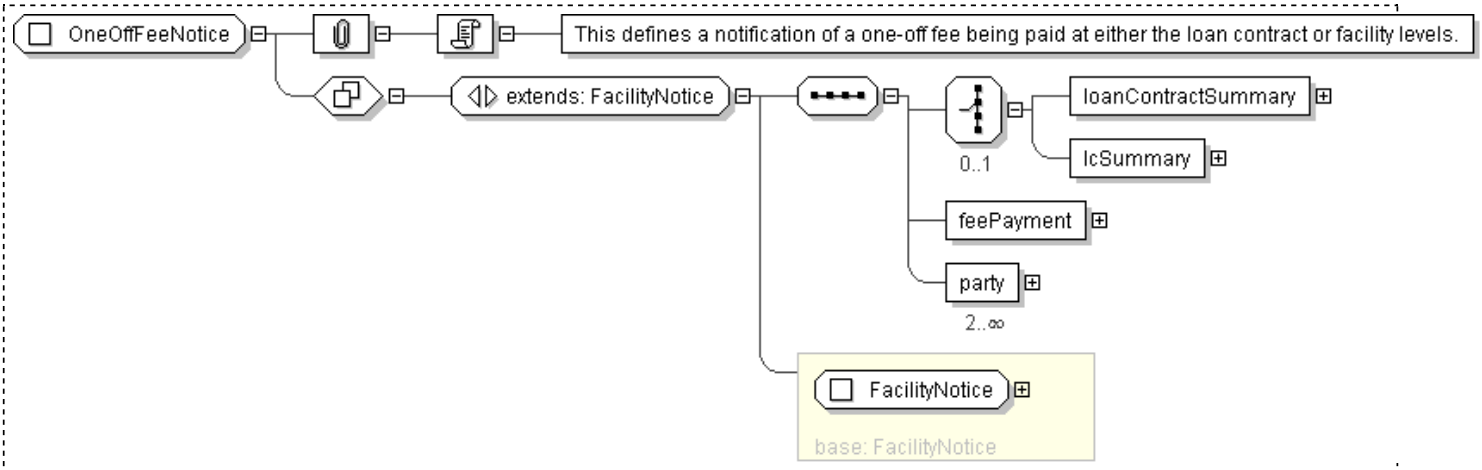
Name	OneOffFeeNotice
Abstract	no
Documentation	This defines a notification of a one-off fee being paid at either the loan contract or facility levels.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify
 which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the
 FpML schema and indicates the build number of the schema file. Every time FpML publishes a
 change to the schema, validation rules, or examples within a version (e.g., version 4.2) the
 actual build number is incremented. If no changes have been made between releases within a
 version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
 same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and
   loan contract levels. This position information is from the message sender\'s viewpoint as of
   the date of the associated notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business
   event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional
   information.'

   Start Choice [0..1]
     <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
     'A basic set of fields used to uniquely identify the loan contract.'
```

```
    <lcSummary> LcSummary </lcSummary> [1]
    'A basic set of fields used to uniquely identify a letter of credit.'

  End Choice
  <feePayment> OneOffFeePayment </feePayment> [1]
  'A representation of the one-off payment.'

  <party> Party </party> [2..*]
  'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OneOffFeeNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
          <xsd:element name="lcSummary" type=" LcSummary "/>
        </xsd:choice>
        <xsd:element name="feePayment" type=" OneOffFeePayment "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OneOffFeePayment

[Table of contents]

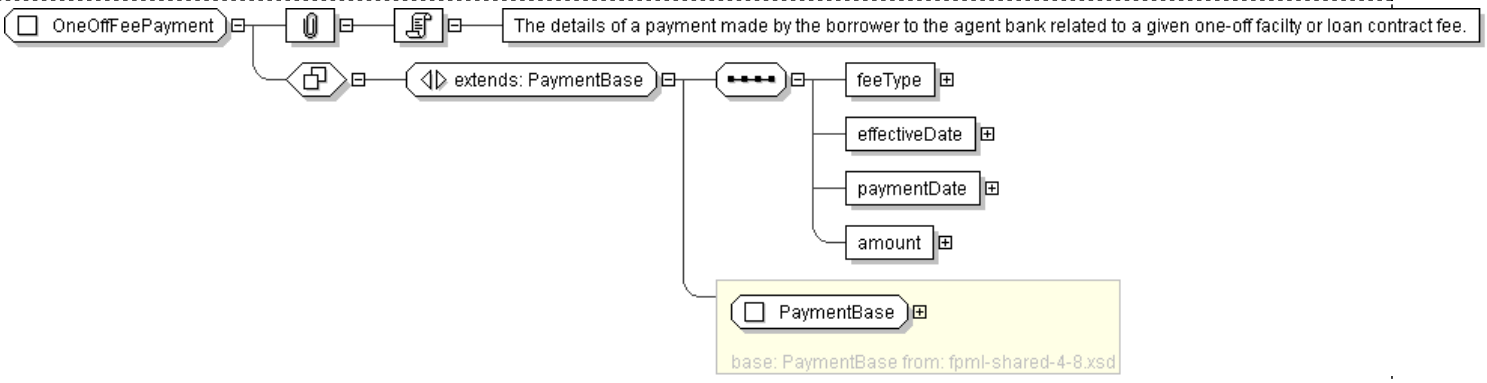
Super-types:	PaymentBase < OneOffFeePayment (by extension)
Sub-types:	None

Name	OneOffFeePayment
Used by (from the same schema document)	Complex Type OneOffFeeNotice
Abstract	no
Documentation	The details of a payment made by the borrower to the agent bank related to a given one-off facilty or loan contract fee.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <feeType> OneOffFeeTypeEnum </feeType> [1]
   'Describes the type of the one-off fee.'

   <effectiveDate> xsd:date </effectiveDate> [1]
   'The date on which the fee is due. It is an actual (adjusted) date.'

   <paymentDate> xsd:date </paymentDate> [1]
   'The date on which the fee is paid by the borrower to the agent bank. This is an actual (adjusted) date.'

   <amount> ParticipationAmount </amount> [1]
   'The global amount describes the amount paid by the borrower to the agent bank whereas the share amount
   describes the amount paid by the agent bank to the lender.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OneOffFeePayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="feeType" type=" OneOffFeeTypeEnum "/>
        <xsd:element name="effectiveDate" type=" xsd:date "/>
        <xsd:element name="paymentDate" type=" xsd:date "/>
        <xsd:element name="amount" type=" ParticipationAmount "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OnGoingFeeNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < OnGoingFeeNotice (by extension)
Sub-types:	None

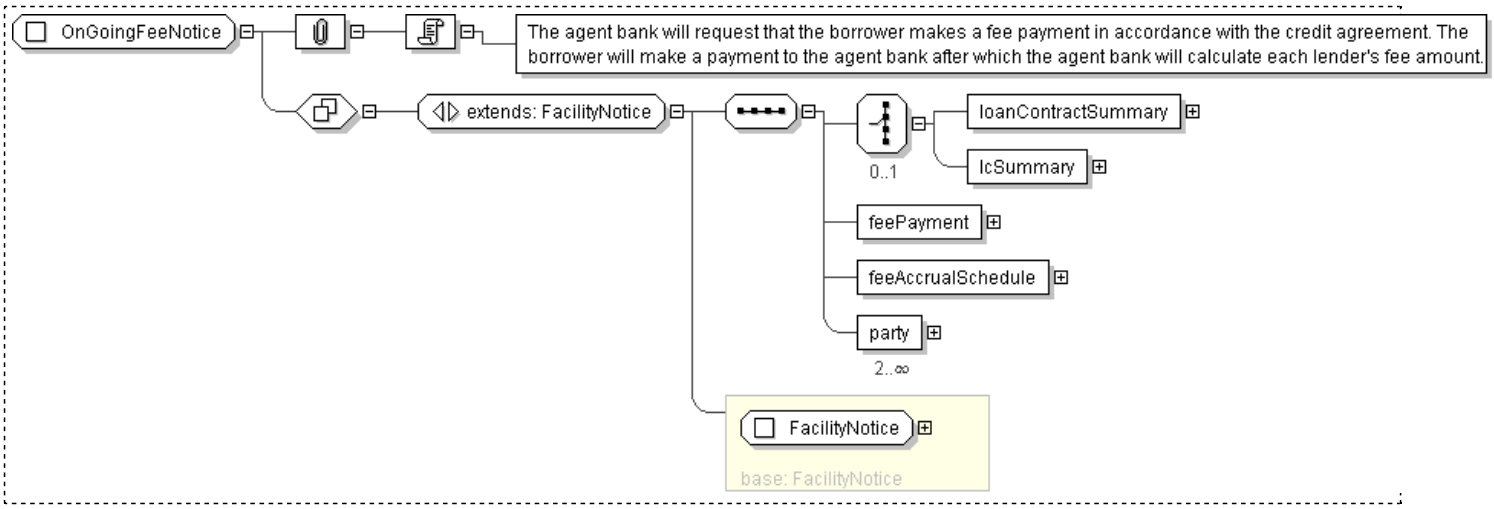
Name	OnGoingFeeNotice
Abstract	no
Documentation	The agent bank will request that the borrower makes a fee payment in accordance with the credit agreement. The borrower will make a payment to the agent bank after which the agent bank will calculate each lender's fee amount.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
 actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan contract
   levels. This position information is from the message sender\'s viewpoint as of the date of the associated
   notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional information.'

   Start Choice [0..1]
      <loanContractSummary> LoanContractSummary </loanContractSummary> [1]
      'A basic set of fields used to uniquely identify the loan contract.'

      <lcSummary> LcSummary </lcSummary> [1]
      'A basic set of fields used to uniquely identify a letter of credit.'

   End Choice
   <feePayment> OnGoingFeePayment </feePayment> [1]
   'A representation of the on-going fee payment.'

   <feeAccrualSchedule> FeeAccrualSchedule </feeAccrualSchedule> [1]
   'The details of the underlying elements that effect the calculation of a fee accrual.'

   <party> Party </party> [2..*]
```

```
   'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OnGoingFeeNotice">
  <xsd:complexContent>
    <xsd:extension base=" FacilityNotice ">
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="loanContractSummary" type=" LoanContractSummary "/>
          <xsd:element name="lcSummary" type=" LcSummary "/>
        </xsd:choice>
        <xsd:element name="feePayment" type=" OnGoingFeePayment "/>
        <xsd:element name="feeAccrualSchedule" type=" FeeAccrualSchedule "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OnGoingFeePayment

[Table of contents]

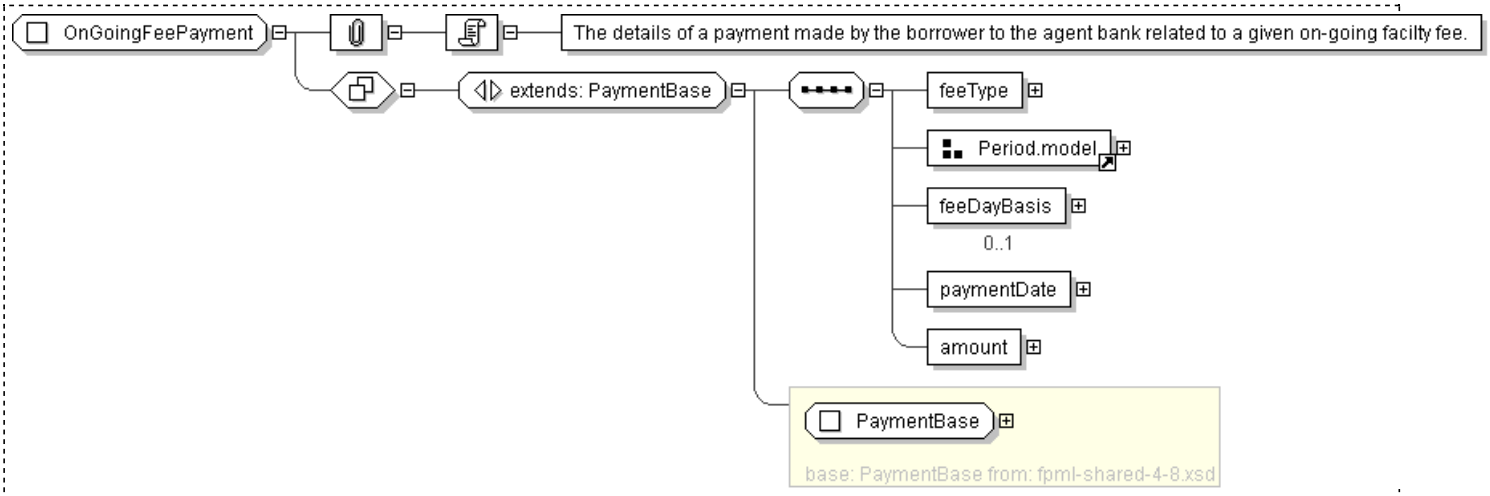
Super-types:	PaymentBase < OnGoingFeePayment (by extension)
Sub-types:	None

Name	OnGoingFeePayment
Used by (from the same schema document)	Complex Type OnGoingFeeNotice
Abstract	no
Documentation	The details of a payment made by the borrower to the agent bank related to a given on-going facilty fee.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <feeType> OnGoingFeeTypeEnum </feeType> [1]
   'Describes the type of the on-going fee.'

   <startDate> xsd:date </startDate> [1]
   'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
   'Date on which this period ends.'

   <feeDayBasis> DayCountFraction </feeDayBasis> [0..1]
   'The day count basis for the fee calculation period.'

   <paymentDate> xsd:date </paymentDate> [1]
   'The day on which the fee is paid by the borrower to the agent bank. This is an actual (adjusted)
   date.'

   <amount> ParticipationAmount </amount> [1]
   'The global amount describes the amount paid by the borrower to the agent bank whereas the share
   amount describes the amount paid by the agent bank to the lender.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OnGoingFeePayment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:element name="feeType" type=" OnGoingFeeTypeEnum "/>
        <xsd:group ref=" Period.model "/>
        <xsd:element name="feeDayBasis" type=" DayCountFraction " minOccurs="0"/>
        <xsd:element name="paymentDate" type=" xsd:date "/>
        <xsd:element name="amount" type=" ParticipationAmount "/>
      </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OnGoingFeeRateChange

[Table of contents]

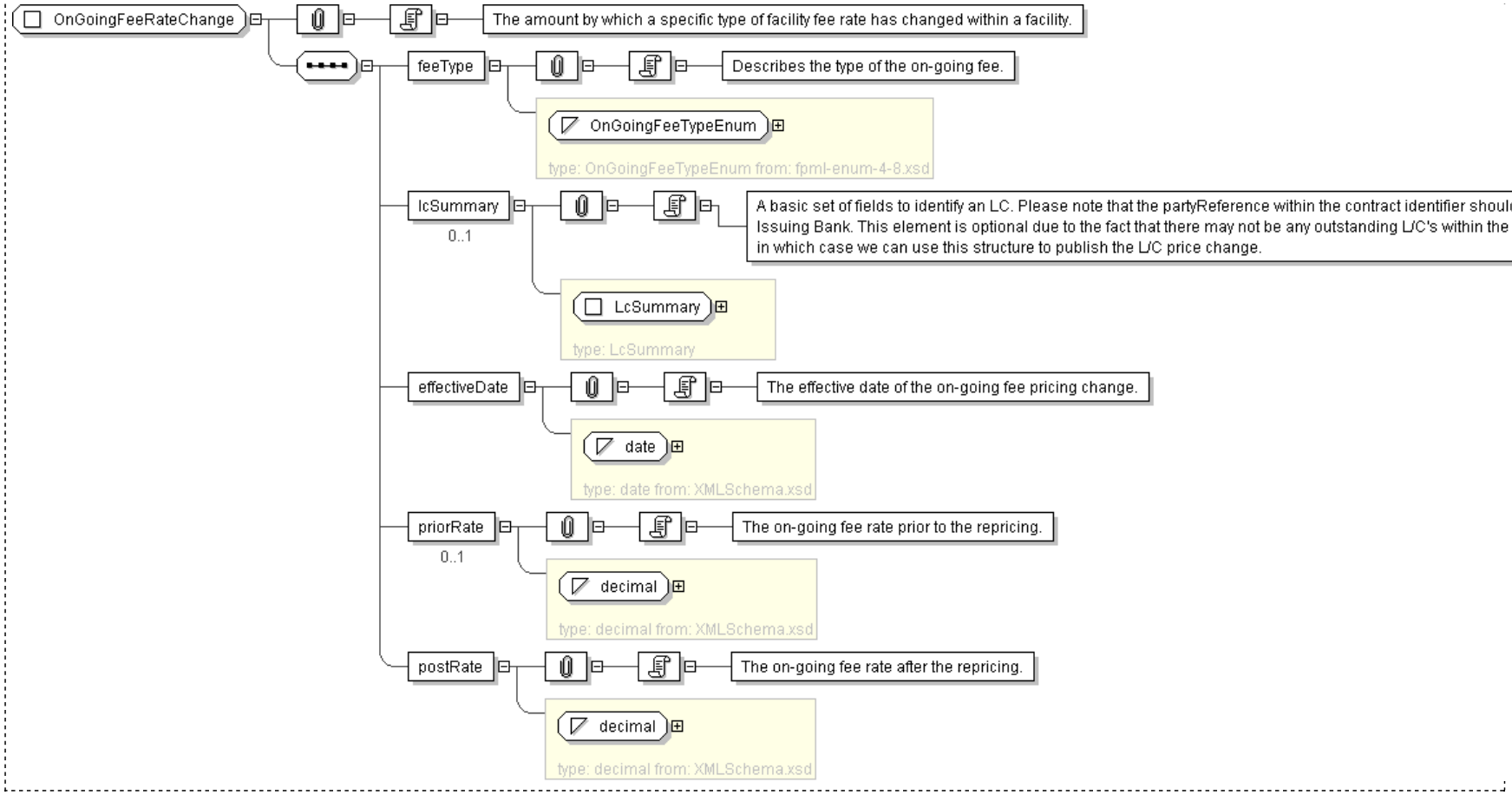
Super-types:	None
Sub-types:	None

Name	OnGoingFeeRateChange
Used by (from the same schema document)	Complex Type PricingChangeNotice , Complex Type PricingChangeNotice
Abstract	no
Documentation	The amount by which a specific type of facility fee rate has changed within a facility.

XML Instance Representation

```
<...>
  <feeType> OnGoingFeeTypeEnum </feeType> [1]
  'Describes the type of the on-going fee.'

  <lcSummary> LcSummary </lcSummary> [0..1]
  'A basic set of fields to identify an LC. Please note that the partyReference within the contract identifier should refer to the Issuing
  Bank. This element is optional due to the fact that there may not be any outstanding L/C\'s within the given facility, in which case we
  can use this structure to publish the L/C price change.'

  <effectiveDate> xsd:date </effectiveDate> [1]
  'The effective date of the on-going fee pricing change.'

  <priorRate> xsd:decimal </priorRate> [0..1]
  'The on-going fee rate prior to the repricing.'

  <postRate> xsd:decimal </postRate> [1]
  'The on-going fee rate after the repricing.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OnGoingFeeRateChange">
  <xsd:sequence>
    <xsd:element name="feeType" type=" OnGoingFeeTypeEnum "/>
    <xsd:element name="lcSummary" type=" LcSummary " minOccurs="0"/>
    <xsd:element name="effectiveDate" type=" xsd:date "/>
    <xsd:element name="priorRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="postRate" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ParticipationAmount

[Table of contents]

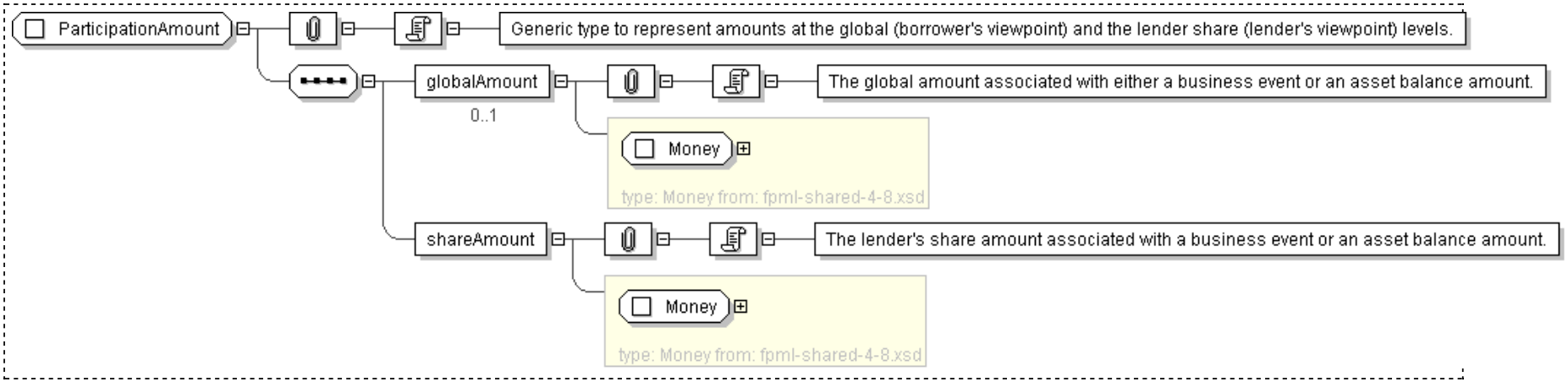
Super-types:	None
Sub-types:	None

Name	ParticipationAmount
Used by (from the same schema document)	Complex Type FacilityCommitmentPosition , Complex Type FacilityCommitmentPosition , Complex Type FacilityRepayment , Complex Type FeeAccrualPeriod , Complex Type InterestPayment , Complex Type InterestRatePeriod , Complex Type LcBalanceNotice , Complex Type LcBalanceNotice , Complex Type LcBalanceNotice , Complex Type LcPosition , Complex Type LcPosition , Complex Type LcTerminationNotice , Complex Type LenderPositionPeriod , Complex Type LoanContract , Complex Type LoanContractPosition , Complex Type LoanContractPosition , Complex Type LoanContractRepayment , Complex Type LoanContractSummary , Complex Type OneOffFeePayment , Complex Type OnGoingFeePayment
Abstract	no
Documentation	Generic type to represent amounts at the global (borrower's viewpoint) and the lender share (lender's viewpoint) levels.

XML Instance Representation

```
<...>
  <globalAmount> Money </globalAmount> [0..1]
  'The global amount associated with either a business event or an asset balance amount.'

  <shareAmount> Money </shareAmount> [1]
  'The lender\'s share amount associated with a business event or an asset balance amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ParticipationAmount">
  <xsd:sequence>
    <xsd:element name="globalAmount" type=" Money " minOccurs="0"/>
    <xsd:element name="shareAmount" type=" Money "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PikPeriod

[Table of contents]

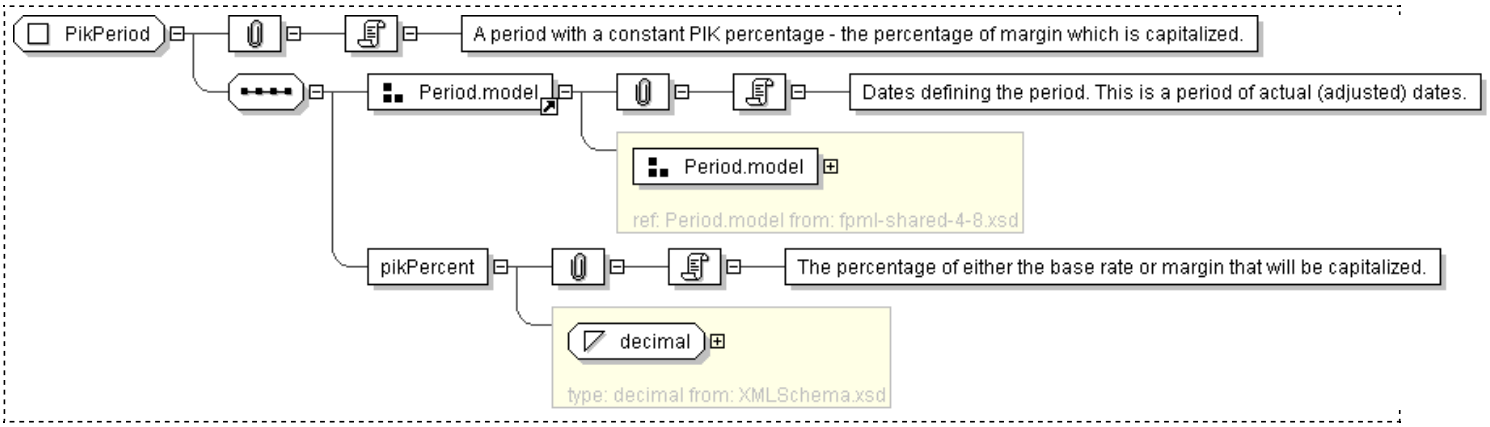
Super-types:	None
Sub-types:	None

Name	PikPeriod
Used by (from the same schema document)	Complex Type InterestAccrualSchedule
Abstract	no
Documentation	A period with a constant PIK percentage - the percentage of margin which is capitalized.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
   'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
   'Date on which this period ends.'

   <pikPercent> xsd:decimal </pikPercent> [1]
   'The percentage of either the base rate or margin that will be capitalized.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PikPeriod">
   <xsd:sequence>
      <xsd:group ref=" Period.model "/>
      <xsd:element name="pikPercent" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingChangeNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < PricingChangeNotice (by extension)
Sub-types:	None

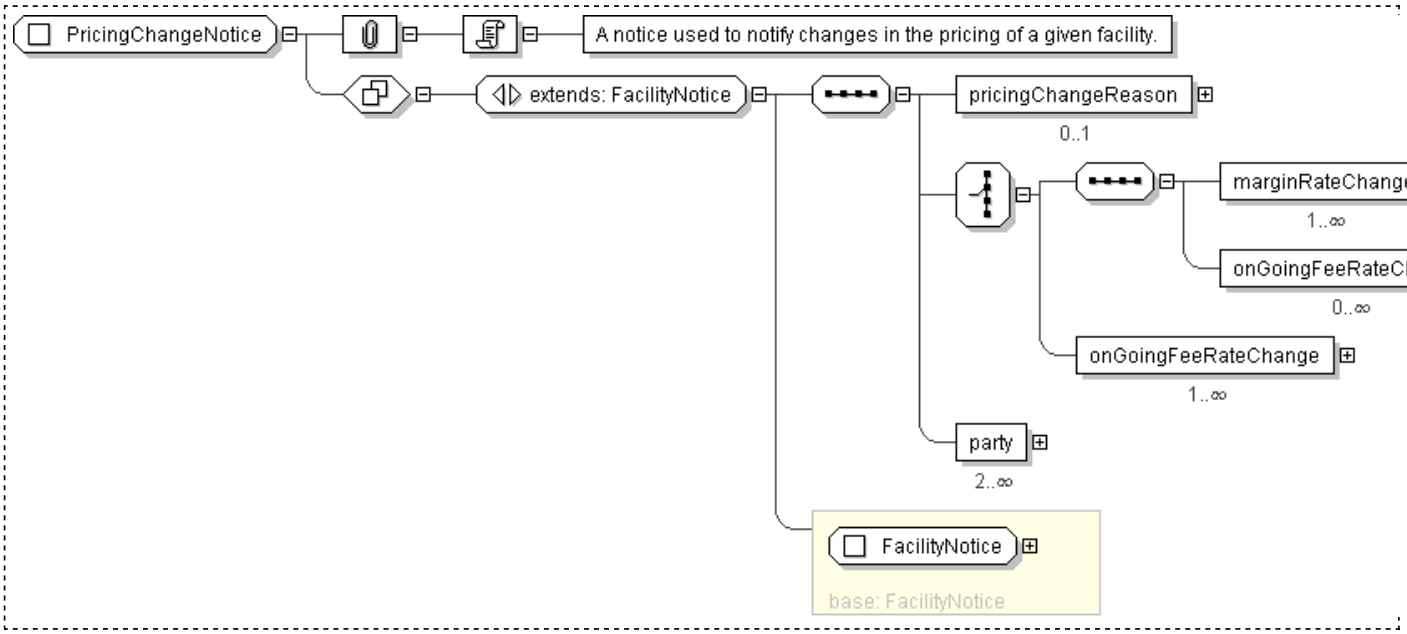
Name	PricingChangeNotice
Abstract	no
Documentation	A notice used to notify changes in the pricing of a given facility.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'})
[1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number
 is incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
   'A structure which contains the position being held by the lender on both the facility and loan
   contract levels. This position information is from the message sender\'s viewpoint as of the date
   of the associated notice.'

   <exceptionFlag> xsd:boolean </exceptionFlag> [0..1]
   'A flag which can be set by the message sender in order to signify an exceptional business
   event.'

   <comments> xsd:string </comments> [0..1]
   'A free-form, manually entered field which will be used by users directly for additional
   information.'

   <pricingChangeReason> PricingChangeReason </pricingChangeReason> [0..1]
   'Denotes the reason a facility has had to be re-priced.'

   Start Choice [1]
      <marginRateChange> MarginRateChange </marginRateChange> [1..*]
      'The amount by which the margin rate has changed within a facility.'
```

```
      <onGoingFeeRateChange> OnGoingFeeRateChange </onGoingFeeRateChange> [0..*]
      ''

      <onGoingFeeRateChange> OnGoingFeeRateChange </onGoingFeeRateChange> [1..*]
      ''

   End Choice
   <party> Party </party> [2..*]
   'The parties involved with the associated transaction.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingChangeNotice">
   <xsd:complexContent>
      <xsd:extension base=" FacilityNotice ">
         <xsd:sequence>
            <xsd:element name="pricingChangeReason" type=" PricingChangeReason " minOccurs="0"/>
            <xsd:choice>
               <xsd:sequence>
                  <xsd:element name="marginRateChange" type=" MarginRateChange " maxOccurs="unbounded"/>
                  <xsd:element name="onGoingFeeRateChange" type=" OnGoingFeeRateChange " minOccurs="0"
                  maxOccurs="unbounded"/>
               </xsd:sequence>
               <xsd:element name="onGoingFeeRateChange" type=" OnGoingFeeRateChange "
               maxOccurs="unbounded"/>
            </xsd:choice>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingChangeReason

[Table of contents]

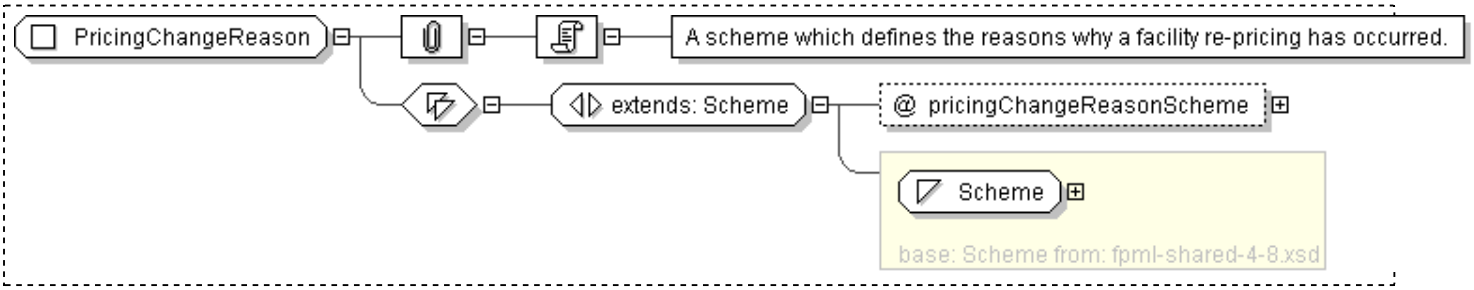
Super-types:	Scheme < PricingChangeReason (by extension)
Sub-types:	None

Name	PricingChangeReason
Used by (from the same schema document)	Complex Type PricingChangeNotice
Abstract	no
Documentation	A scheme which defines the reasons why a facility re-pricing has occurred.

XML Instance Representation

```
<...
 pricingChangeReasonScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingChangeReason">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="pricingChangeReasonScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RatePeriod

[Table of contents]

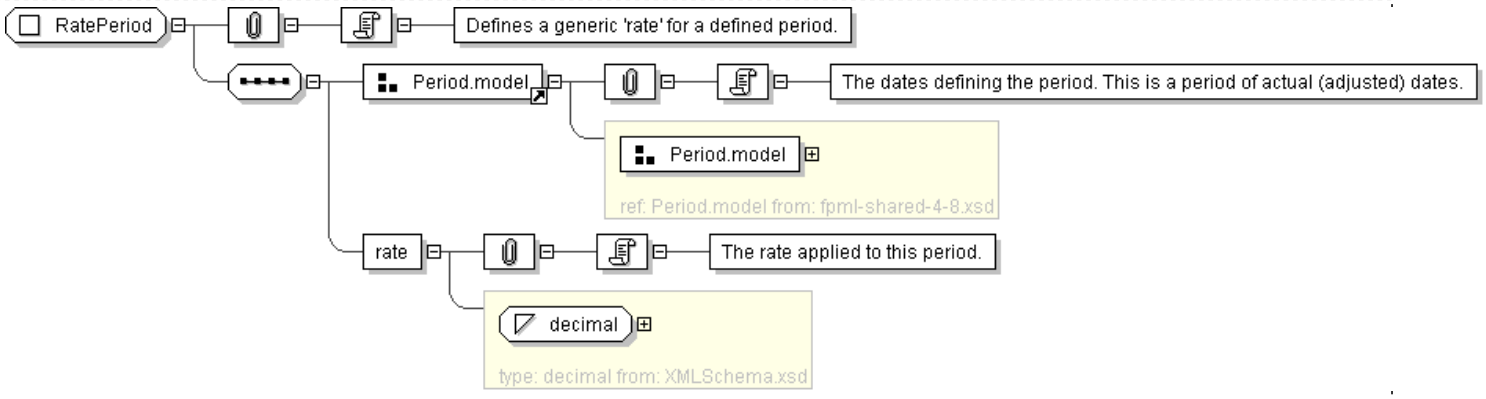
Super-types:	None
Sub-types:	None

Name	RatePeriod
Used by (from the same schema document)	Complex Type FeeAccrualSchedule
Abstract	no
Documentation	Defines a generic 'rate' for a defined period.

XML Instance Representation

```
<...>
  <startDate> xsd:date </startDate> [1]
  'Date on which this period begins.'

  <endDate> xsd:date </endDate> [1]
  'Date on which this period ends.'

  <rate> xsd:decimal </rate> [1]
  'The rate applied to this period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RatePeriod">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:element name="rate" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Repayment

[Table of contents]

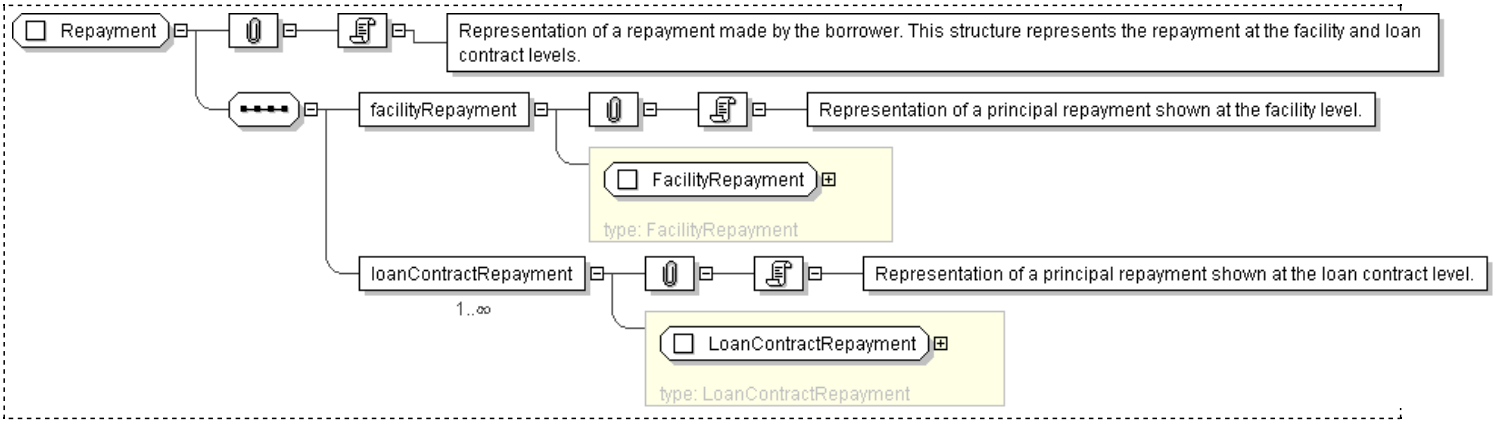
Super-types:	None
Sub-types:	None

Name	Repayment
Used by (from the same schema document)	Complex Type MaturingLoanContracts , Complex Type RepaymentNotice
Abstract	no
Documentation	Representation of a repayment made by the borrower. This structure represents the repayment at the facility and loan contract levels.

XML Instance Representation

```
<...>
  <facilityRepayment> FacilityRepayment </facilityRepayment> [1]
  'Representation of a principal repayment shown at the facility level.'

  <loanContractRepayment> LoanContractRepayment </loanContractRepayment> [1..*]
  'Representation of a principal repayment shown at the loan contract level.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Repayment">
  <xsd:sequence>
    <xsd:element name="facilityRepayment" type=" FacilityRepayment "/>
    <xsd:element name="loanContractRepayment" type=" LoanContractRepayment " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RepaymentNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < RepaymentNotice (by extension)
Sub-types:	None

Name	RepaymentNotice
Abstract	no
Documentation	A notice describing a principal repayment to be made by the borrower.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'
```

<facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]

'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]

'A flag which can be set by the message sender in order to signify an exceptional business event.'

<comments> xsd:string </comments> [0..1]

'A free-form, manually entered field which will be used by users directly for additional information.'

<repayment> Repayment </repayment> [1]

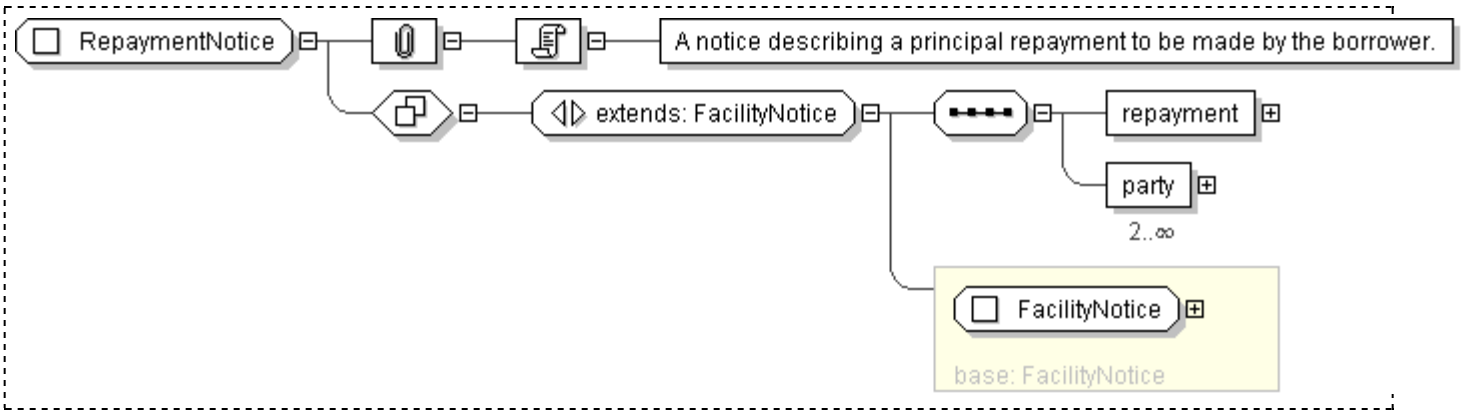
'Representation of a repayment made by the borrower. This structure represents the repayment at the facility and loan contract levels.'

<party> Party </party> [2..*]

'The parties involved with the associated transaction.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="RepaymentNotice">
   <xsd:complexContent>
      <xsd:extension base=" FacilityNotice ">
         <xsd:sequence>
            <xsd:element name="repayment" type=" Repayment "/>
            <xsd:element name="party" type=" Party " minOccurs="2"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RolloverNotice

[Table of contents]

Super-types:	NotificationMessage < FacilityNotice (by extension) < RolloverNotice (by extension)
Sub-types:	None

Name	RolloverNotice
Abstract	no
Documentation	A rollover notice used for both simple as well as complex rollover situations.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <noticeDate> xsd:date </noticeDate> [1]
   'The date on which the notice was generated.'

   <eventId> EventId </eventId> [0..1]
   'The identifier that defines the business event which requires the creation of a notice.'

   <agentBankPartyReference> PartyReference </agentBankPartyReference> [0..1]
   'A reference to the agent bank for the given deal.'

   <borrowerPartyReference> PartyReference </borrowerPartyReference> [0..1]
   'A reference to the borrower against the associated loan contract(s).'

   <lenderPartyReference> PartyReference </lenderPartyReference> [0..1]
   'A reference to the lender(s) associated with the associated loan contract(s).'

   <dealSummary> DealSummary </dealSummary> [1]
   'A data structure which contains the identifying characteristics of the given deal.'

   <facilitySummary> FacilitySummary </facilitySummary> [1]
   'A data structure which contains the identifying characteristics of the given facility.'

   <facilityCommitmentPosition> FacilityCommitmentPosition </facilityCommitmentPosition> [0..1]
```

'A structure which contains the position being held by the lender on both the facility and loan contract levels. This position information is from the message sender\'s viewpoint as of the date of the associated notice.'

<exceptionFlag> xsd:boolean </exceptionFlag> [0..1]

'A flag which can be set by the message sender in order to signify an exceptional business event.'

<comments> xsd:string </comments> [0..1]

'A free-form, manually entered field which will be used by users directly for additional information.'

<maturingLoanContracts> MaturingLoanContracts </maturingLoanContracts> [1]

'Details of all loan contracts due to mature and all corresponding interest payments to be paid for the given rollover event.'

<newLoanContracts> NewLoanContracts </newLoanContracts> [1]

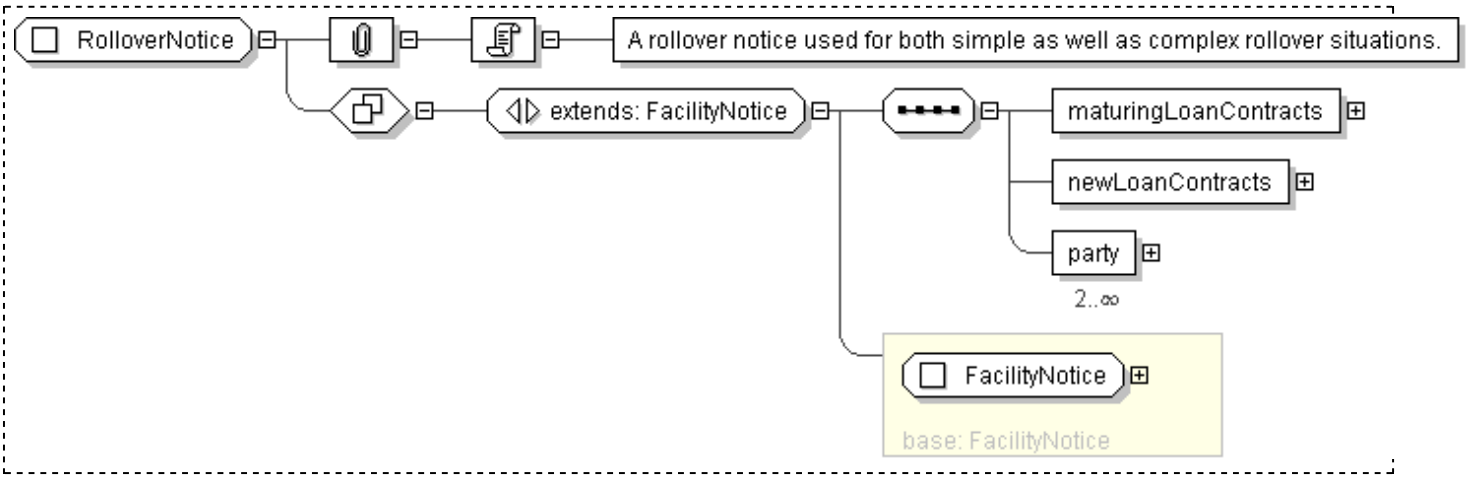
'A list of the new loan contracts to be created as a result of this rollover.'

<party> Party </party> [2..*]

'The parties involved with the associated transaction.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="RolloverNotice">
   <xsd:complexContent>
      <xsd:extension base=" FacilityNotice ">
         <xsd:sequence>
            <xsd:element name="maturingLoanContracts" type=" MaturingLoanContracts "/>
            <xsd:element name="newLoanContracts" type=" NewLoanContracts "/>
            <xsd:element name="party" type=" Party " minOccurs="2"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: NovationMatched
  - Complex Type: TradeAlleged
  - Complex Type: TradeMatched
  - Complex Type: TradeMismatched
  - Complex Type: TradeUnmatched
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema

component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationMatched

[Table of contents]

Super-types:	NovationNotificationMessage < NovationMatched (by extension)
Sub-types:	None

Name	NovationMatched
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="NovationMatched">
   <xsd:complexContent>
      <xsd:extension base=" NovationNotificationMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlleged

[Table of contents]

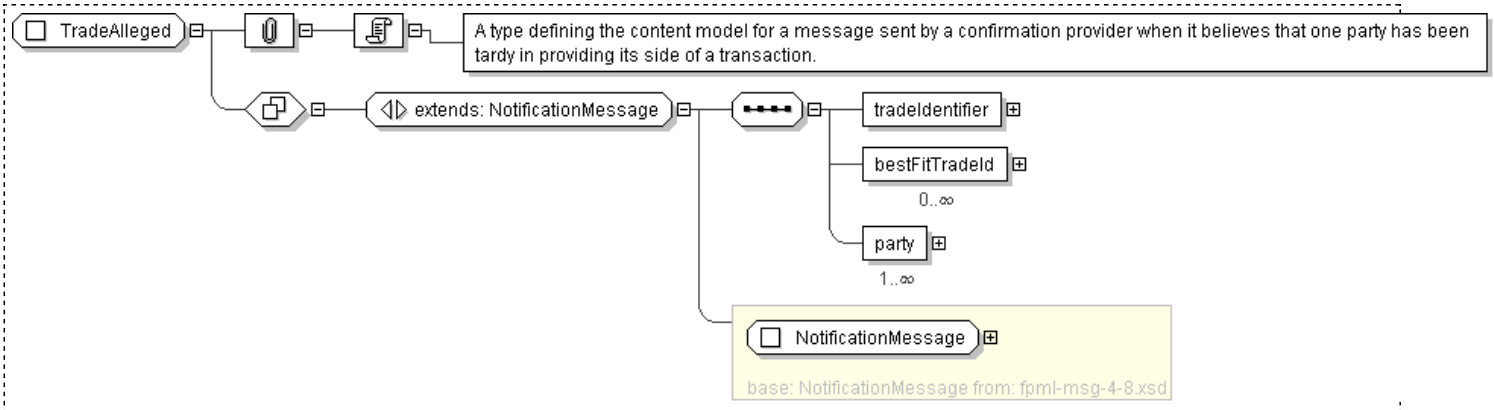
Super-types:	NotificationMessage < TradeAlleged (by extension)
Sub-types:	None

Name	TradeAlleged
Abstract	no
Documentation	A type defining the content model for a message sent by a confirmation provider when it believes that one party has been tardy in providing its side of a transaction.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <bestFitTradeId> TradeIdentifier </bestFitTradeId> [0..*]
   'A trade identifier for a transaction that closely resembles the characteristics of the trade under
   consideration.'

   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term
   of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting
   in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in
   multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlleged">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
        <xsd:element name="bestFitTradeId" type=" TradeIdentifier " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
```

```
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeMatched

[Table of contents]

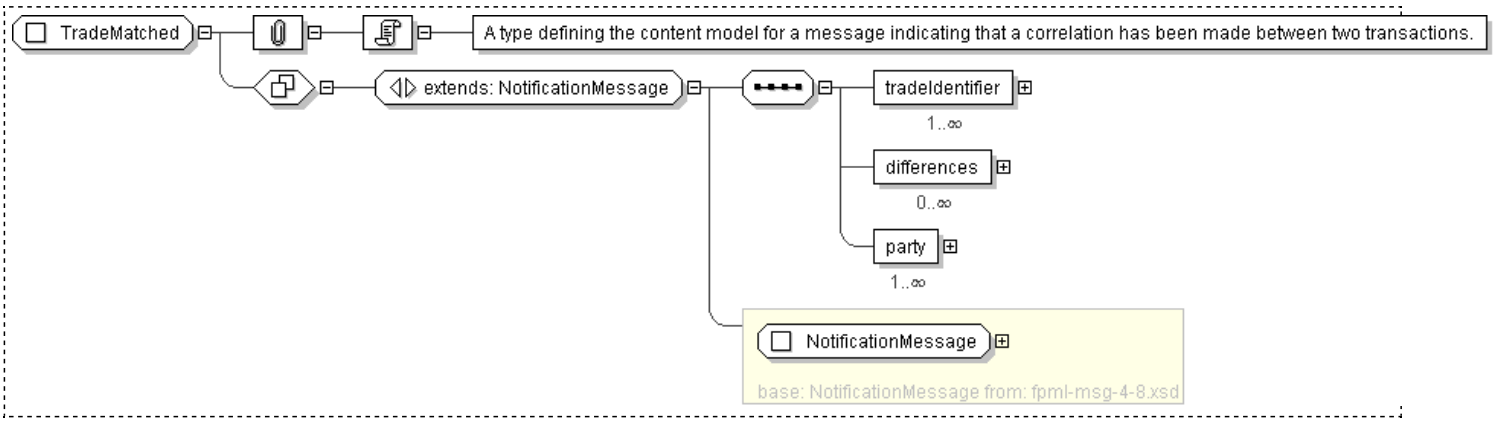
Super-types:	NotificationMessage < TradeMatched (by extension)
Sub-types:	None

Name	TradeMatched
Abstract	no
Documentation	A type defining the content model for a message indicating that a correlation has been made between two transactions.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1..*]
   'An instance of a unique trade identifier.'

   <differences> TradeDifference </differences> [0..*]
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term
   of the trade, but may include other parties involved in, or incidental to, the trade, such as parties
   acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are
   defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeMatched">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier " maxOccurs="unbounded"/>
        <xsd:element name="differences" type=" TradeDifference " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeMismatched

[Table of contents]

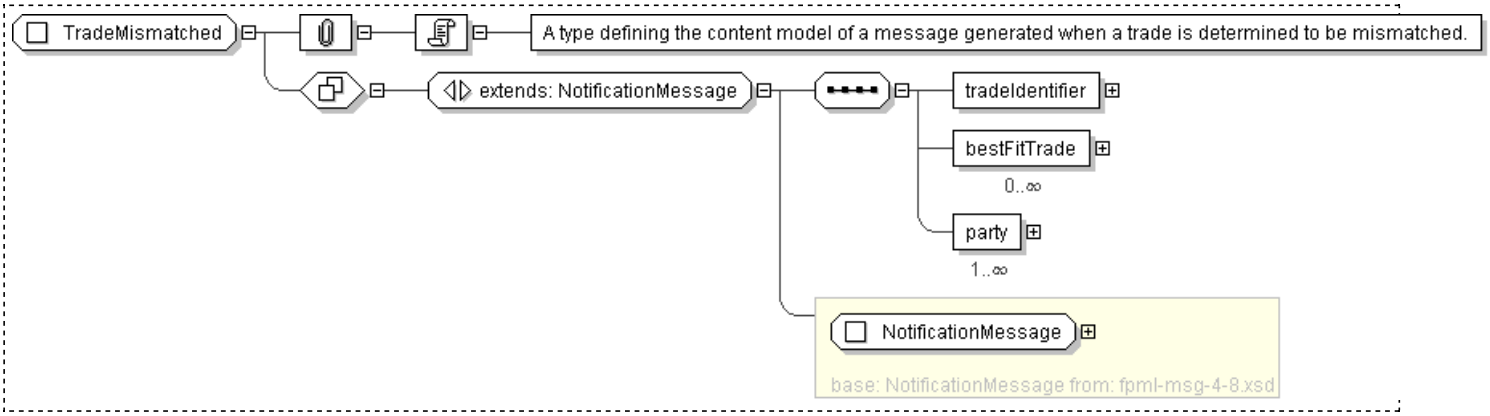
Super-types:	NotificationMessage < TradeMismatched (by extension)
Sub-types:	None

Name	TradeMismatched
Abstract	no
Documentation	A type defining the content model of a message generated when a trade is determined to be mismatched.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <bestFitTrade> BestFitTrade </bestFitTrade> [0..*]
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the
   term of the trade, but may include other parties involved in, or incidental to, the trade, such as
   parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML
   roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeMismatched">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
        <xsd:element name="bestFitTrade" type=" BestFitTrade " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeUnmatched

[Table of contents]

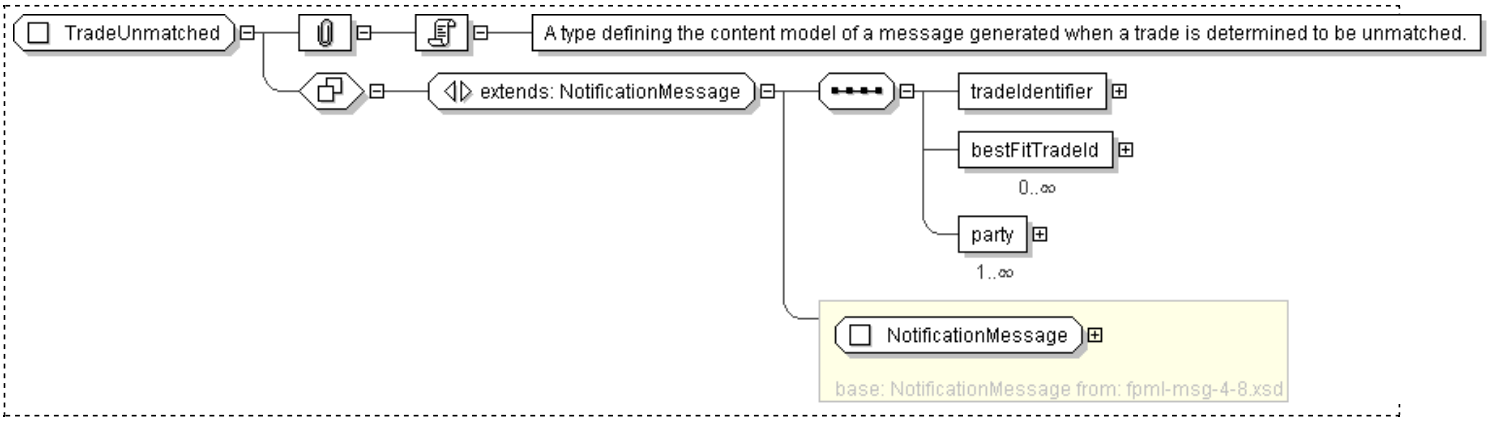
Super-types:	NotificationMessage < TradeUnmatched (by extension)
Sub-types:	None

Name	TradeUnmatched
Abstract	no
Documentation	A type defining the content model of a message generated when a trade is determined to be unmatched.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <bestFitTradeId> TradeIdentifier </bestFitTradeId> [0..*]
   'A trade identifier for a transaction that closely resembles the characteristics of the trade under
   consideration.'

   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during
   the term of the trade, but may include other parties involved in, or incidental to, the trade, such
   as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In
   FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeUnmatched">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
```

```
          <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
          <xsd:element name="bestFitTradeId" type=" TradeIdentifier " minOccurs="0" maxOccurs="unbounded"/>
          <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: creditCurve
  - Element: creditCurveValuation
  - Element: fxCurve
  - Element: fxCurveValuation
  - Element: volatilityMatrixValuation
  - Element: volatilityRepresentation
  - Element: yieldCurve
  - Element: yieldCurveValuation
- Global Definitions
  - Complex Type: CompoundingFrequency
  - Complex Type: CreditCurve
  - Complex Type: CreditCurveValuation
  - Complex Type: DefaultProbabilityCurve
  - Complex Type: ForwardRateCurve
  - Complex Type: FxCurve
  - Complex Type: FxCurveValuation
  - Complex Type: FxRateSet
  - Complex Type: MultiDimensionalPricingData
  - Complex Type: ParametricAdjustment
  - Complex Type: ParametricAdjustmentPoint
  - Complex Type: PricingStructurePoint
  - Complex Type: TermCurve
  - Complex Type: TermPoint
  - Complex Type: VolatilityMatrix
  - Complex Type: VolatilityRepresentation
  - Complex Type: YieldCurve
  - Complex Type: YieldCurveValuation
  - Complex Type: ZeroRateCurve
  - Model Group: BidMidAsk.model
  - Model Group: CreditCurveCharacteristics.model
  - Model Group: FxCurveCharacteristics.model
  - Model Group: RecoveryRate.model
  - Model Group: UnderlyingAssetOrReference.model
  - Model Group: YieldCurveCharacteristics.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	

- This schema includes components from the following schema document(s):
  - fpml-riskdef-4-8.xsd
  - fpml-cd-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-riskdef-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-cd-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress
Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness

Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

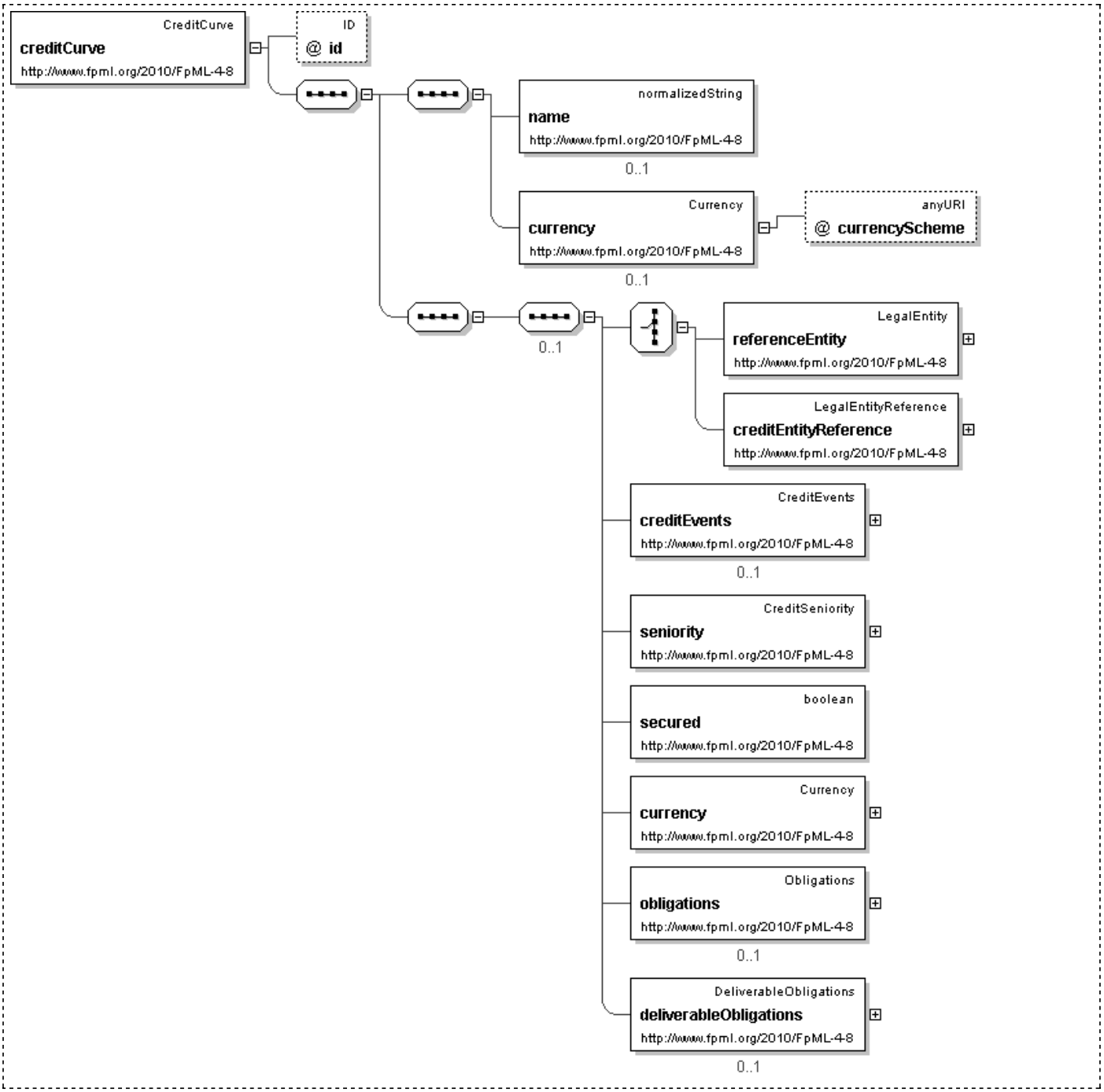
Element: creditCurve

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructure

Name	creditCurve
Type	CreditCurve
Nillable	no
Abstract	no

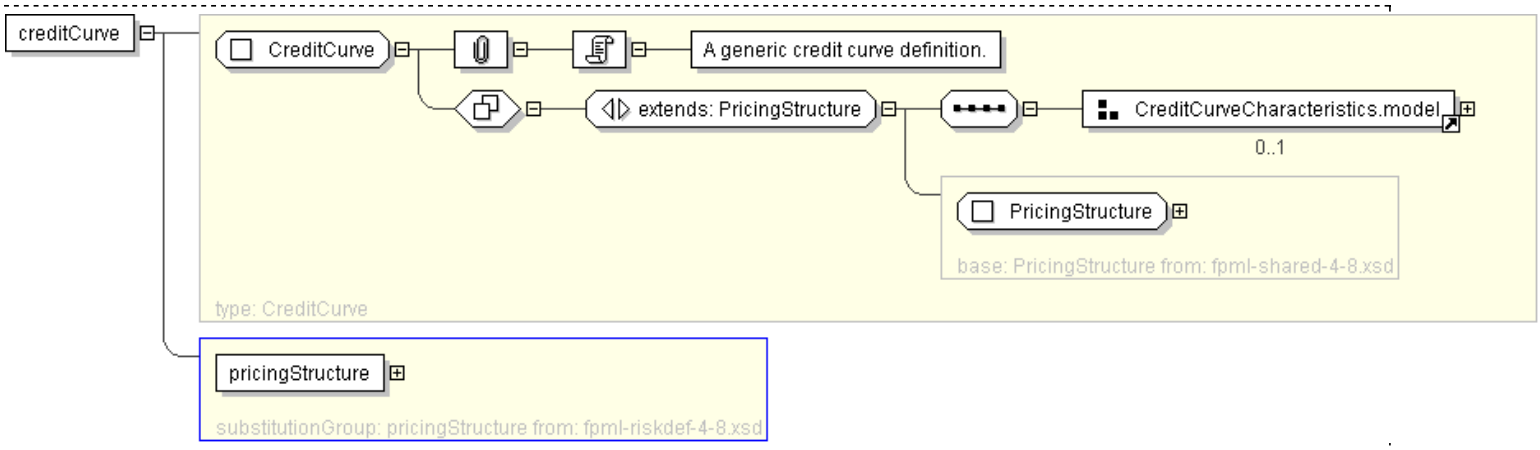
Logical Diagram



XML Instance Representation

```
<creditCurve
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates
   asset class).'

   Start Group: CreditCurveCharacteristics.model [0..1]
      Start Choice [1]
         <referenceEntity> LegalEntity </referenceEntity> [1]
         'The entity for which this is defined.'

         <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
         'An XML reference a credit entity defined elsewhere in the document.'

      End Choice
      <creditEvents> CreditEvents </creditEvents> [0..1]
      'The material credit event.'

      <seniority> CreditSeniority </seniority> [1]
      'The level of seniority of the deliverable obligation.'

      <secured> xsd:boolean </secured> [1]
      'Whether the deliverable obligation is secured or unsecured.'

      <currency> Currency </currency> [1]
      'The currency of denomination of the deliverable obligation.'

      <obligations> Obligations </obligations> [0..1]
      'The underlying obligations of the reference entity on which you are buying or selling
      protection'

      <deliverableObligations> DeliverableObligations </deliverableObligations> [0..1]
      'What sort of obligation may be delivered in the event of the credit event. ISDA 2003 Term:
      Obligation Category/Deliverable Obligation Category'

   End Group: CreditCurveCharacteristics.model
</creditCurve>
```

Diagram



Schema Component Representation

```
<xsd:element name="creditCurve" type=" CreditCurve " substitutionGroup="pricingStructure"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

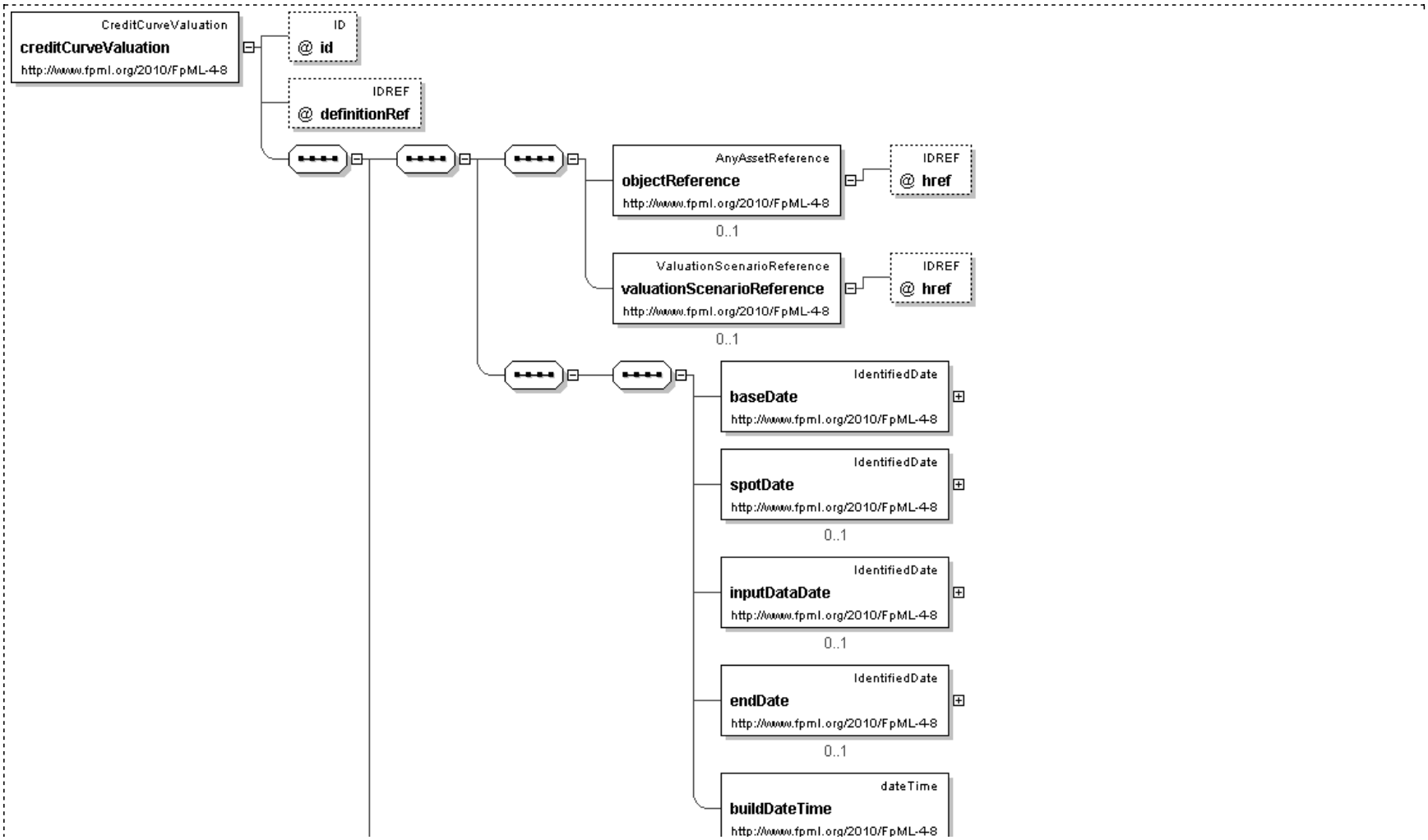
Element: creditCurveValuation

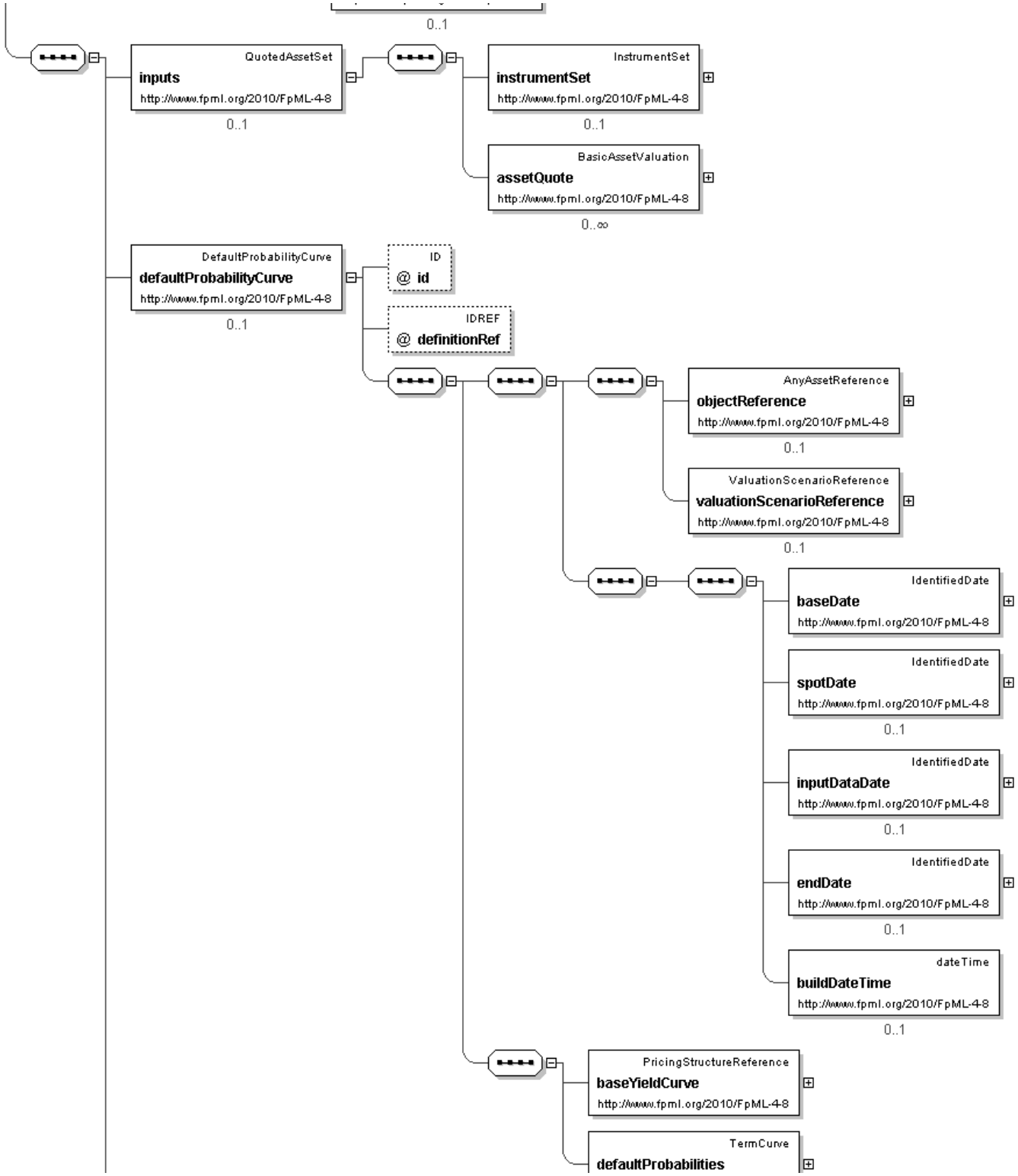
[Table of contents]

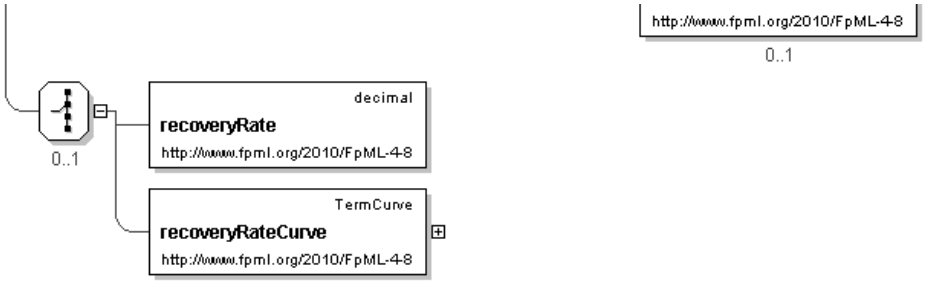
- *This element can be used wherever the following element is referenced:*
  - pricingStructureValuation

Name	creditCurveValuation
Type	CreditCurveValuation
Nillable	no
Abstract	no

Logical Diagram



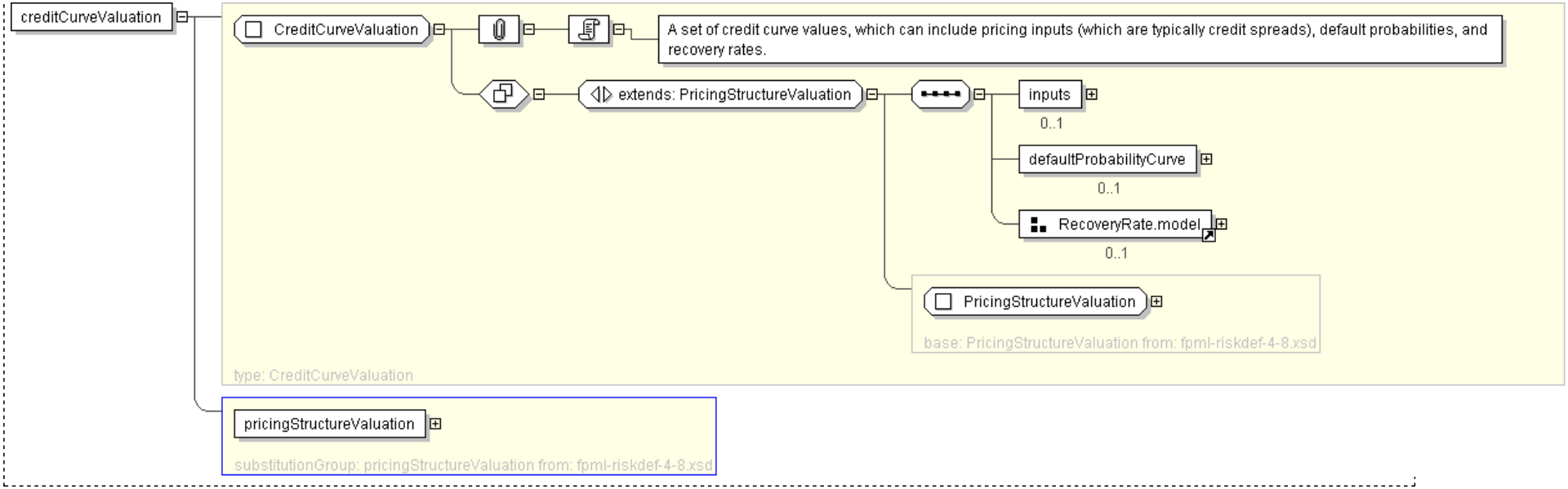

0..1
QuotedAssetSet
inputs
http://www.fpml.org/2010/FpML-4-8
0..1
InstrumentSet
instrumentSet
http://www.fpml.org/2010/FpML-4-8
0..1
BasicAssetValuation
assetQuote
http://www.fpml.org/2010/FpML-4-8
0..∞
DefaultProbabilityCurve
defaultProbabilityCurve
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
IDREF
@ definitionRef
AnyAssetReference
objectReference
http://www.fpml.org/2010/FpML-4-8
0..1
ValuationScenarioReference
valuationScenarioReference
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
baseDate
http://www.fpml.org/2010/FpML-4-8
IdentifiedDate
spotDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
inputDataDate
http://www.fpml.org/2010/FpML-4-8
0..1
IdentifiedDate
endDate
http://www.fpml.org/2010/FpML-4-8
0..1
dateTime
buildDateTime
http://www.fpml.org/2010/FpML-4-8
0..1
PricingStructureReference
baseYieldCurve
http://www.fpml.org/2010/FpML-4-8
TermCurve
defaultProbabilities



XML Instance Representation

```
<creditCurveValuation
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a ValuationSet, this value is
   optional and is defaulted from the ValuationSet. If this value occurs in both places, the lower level value (i.e. the one here)
   overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference between the baseDate
   and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the baseDate, but sometimes
   the pricing input may be \"rolled forward\", in which input data from one date is used to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <inputs> QuotedAssetSet </inputs> [0..1]
   <defaultProbabilityCurve> DefaultProbabilityCurve </defaultProbabilityCurve> [0..1]
   'A curve of default probabilities.'

   Start Group: RecoveryRate.model [0..1]
'A recovery rate value or curve.'


      Start Choice [1]
         <recoveryRate> xsd:decimal </recoveryRate> [1]
         'A single recovery rate, to be used for all terms.'

         <recoveryRateCurve> TermCurve </recoveryRateCurve> [1]
         'A curve of recovery rates, allowing different terms to have different recovery rates.'

      End Choice
   End Group: RecoveryRate.model
</creditCurveValuation>
```

Diagram



Schema Component Representation

```
<xsd:element name="creditCurveValuation" type=" CreditCurveValuation " substitutionGroup="pricingStructureValuation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

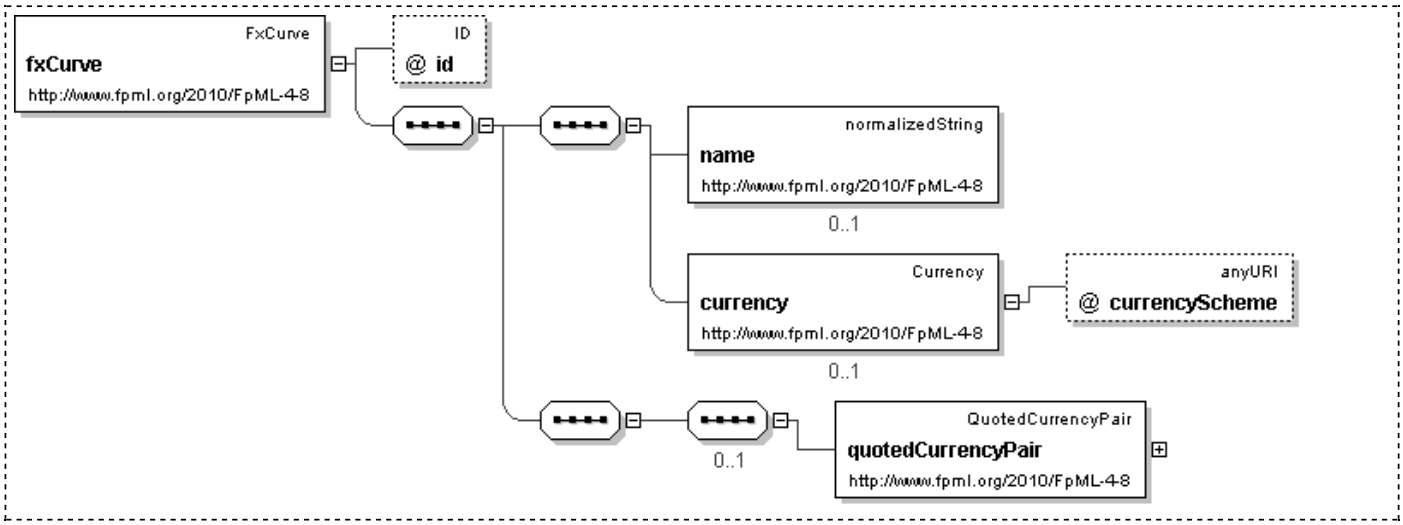
Element: fxCurve

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructure

Name	fxCurve
Type	FxCurve
Nillable	no
Abstract	no

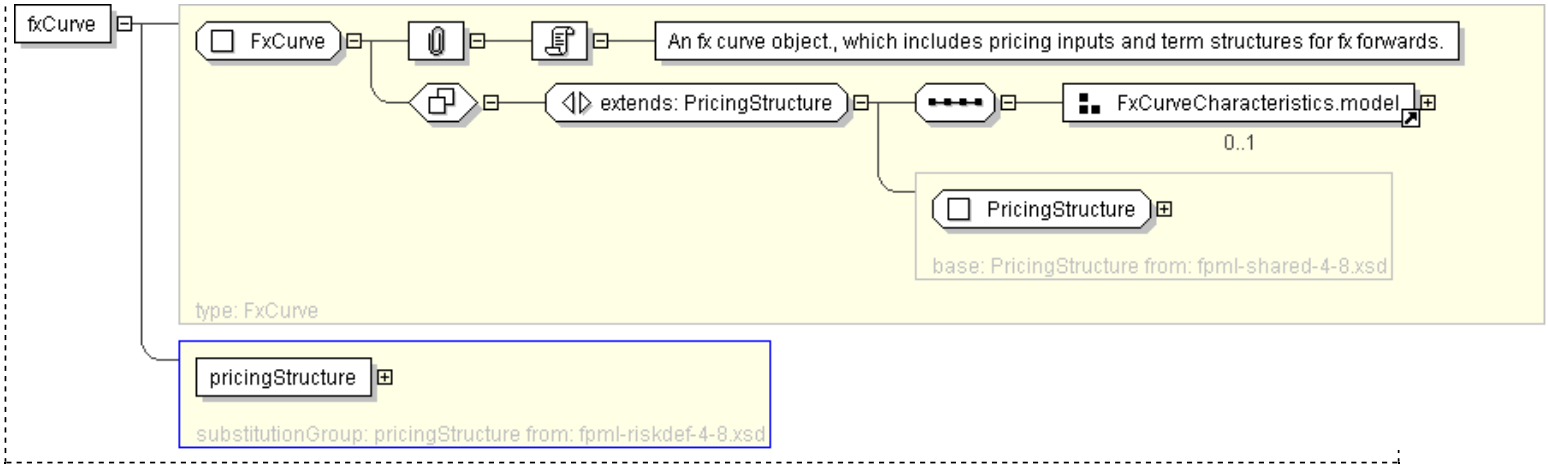
Logical Diagram



XML Instance Representation

```
<fxCurve
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
    'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
    'The currency that the structure is expressed in (this is relevant mostly for the Interes
    Rates asset class).'

   Start Group: FxCurveCharacteristics.model [0..1]
      <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
       'Defines the two currencies for an FX trade and the quotation relationship between the two
       currencies.'

   End Group: FxCurveCharacteristics.model
</fxCurve>
```

Diagram



Schema Component Representation

```
<xsd:element name="fxCurve" type=" FxCurve " substitutionGroup="pricingStructure"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

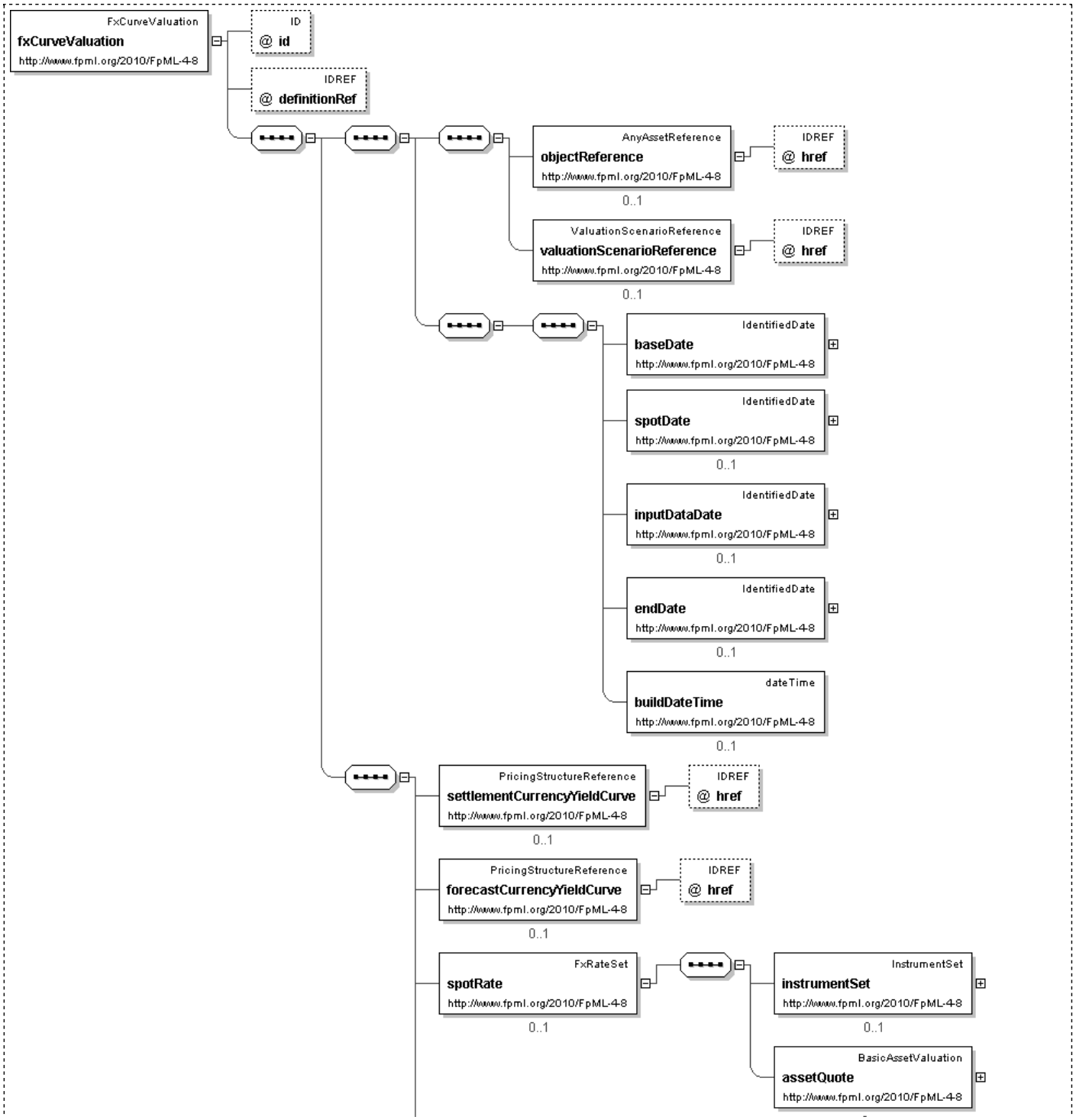
Element: fxCurveValuation

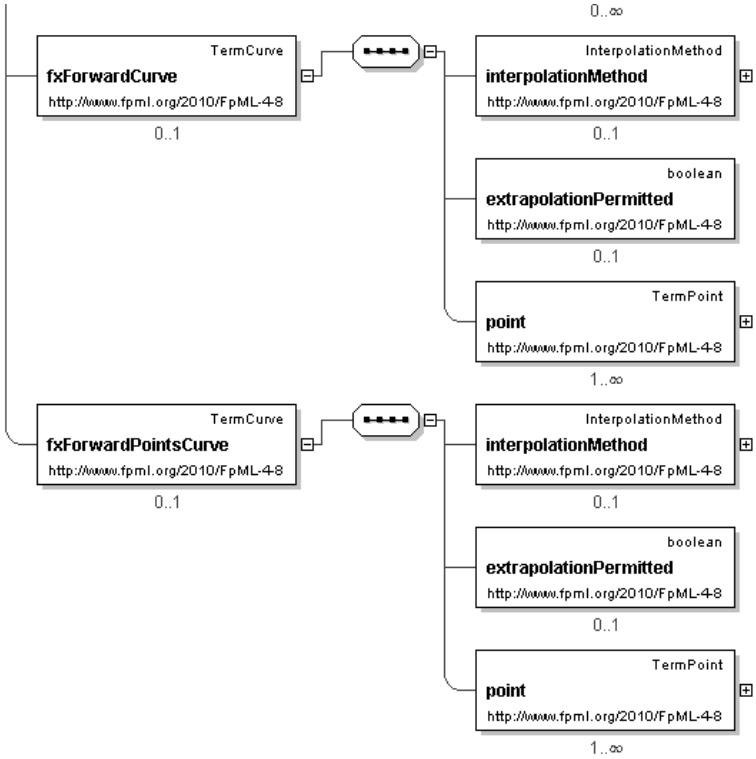
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructureValuation

Name	fxCurveValuation
Type	FxCurveValuation
Nillable	no
Abstract	no

Logical Diagram

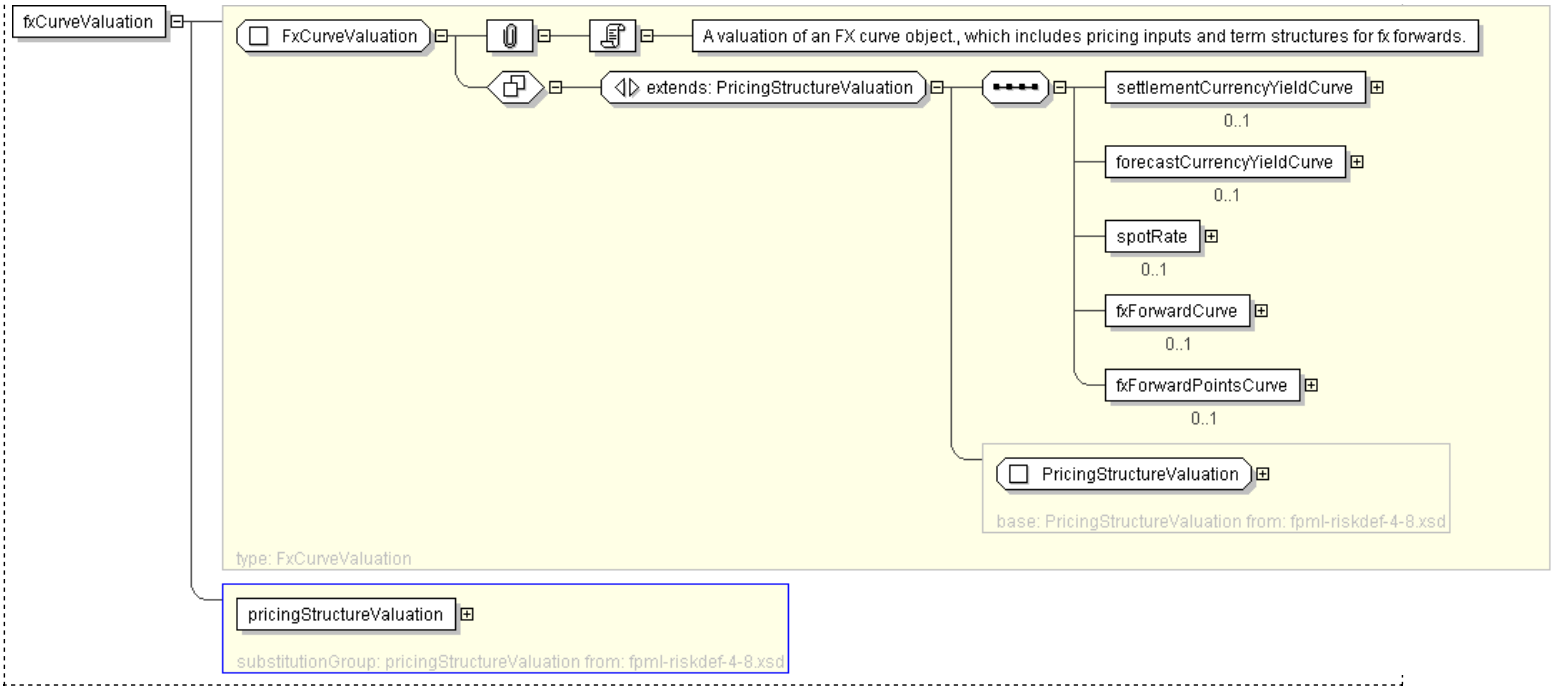




XML Instance Representation

```
<fxCurveValuation
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a
   ValuationSet, this value is optional and is defaulted from the ValuationSet. If this value occurs in both places,
   the lower level value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation
   date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference
   between the baseDate and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the
   baseDate, but sometimes the pricing input may be \"rolled forward\", in which input data from one date is used to
   generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <settlementCurrencyYieldCurve> PricingStructureReference </settlementCurrencyYieldCurve> [0..1]
   <forecastCurrencyYieldCurve> PricingStructureReference </forecastCurrencyYieldCurve> [0..1]
   <spotRate> FxRateSet </spotRate> [0..1]
   <fxForwardCurve> TermCurve </fxForwardCurve> [0..1]
   'A curve of fx forward rates.'

   <fxForwardPointsCurve> TermCurve </fxForwardPointsCurve> [0..1]
   'A curve of fx forward point spreads.'

</fxCurveValuation>
```

Diagram



Schema Component Representation

```
<xsd:element name="fxCurveValuation" type=" FxCurveValuation " substitutionGroup="pricingStructureValuation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

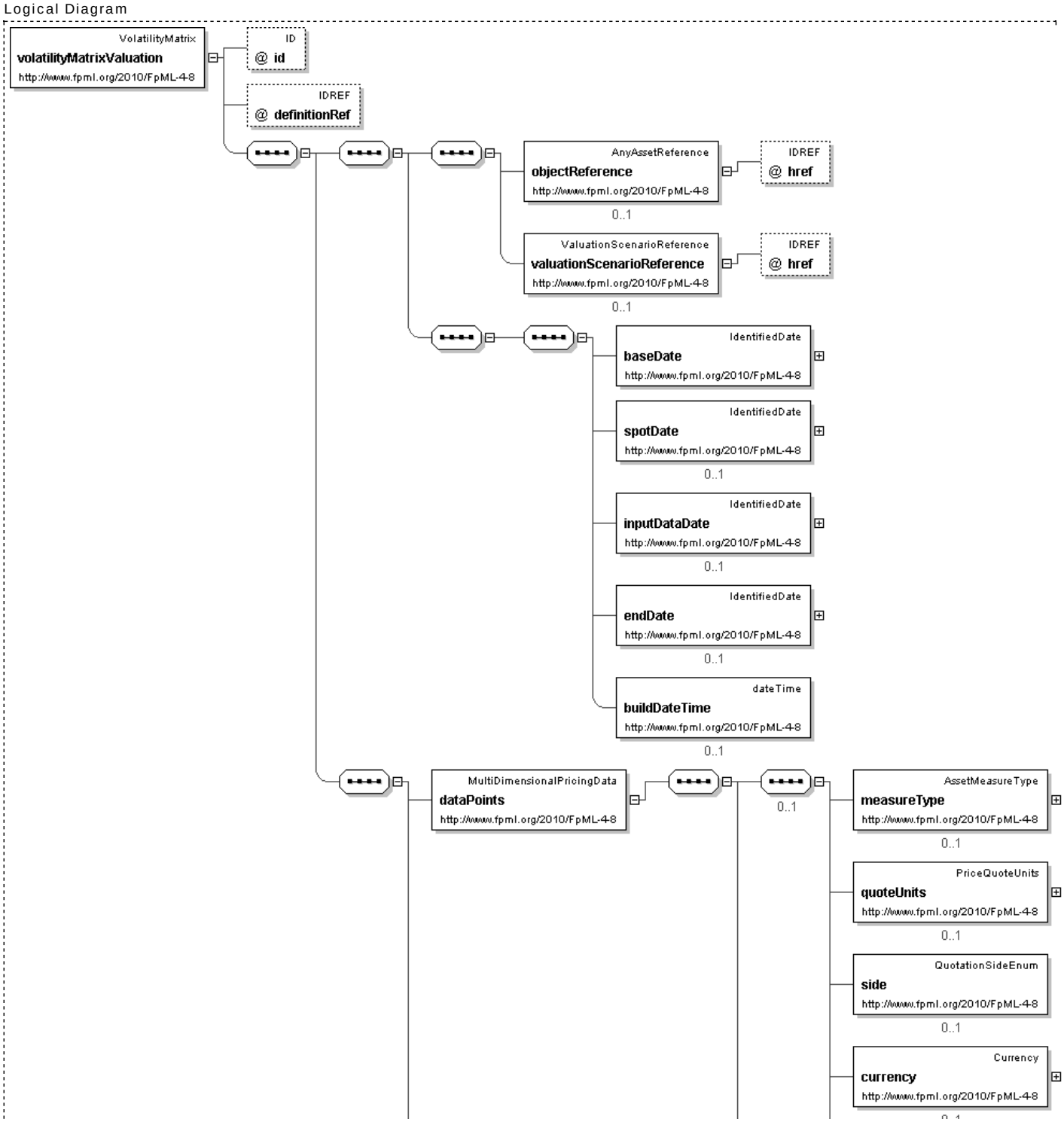
Element: volatilityMatrixValuation

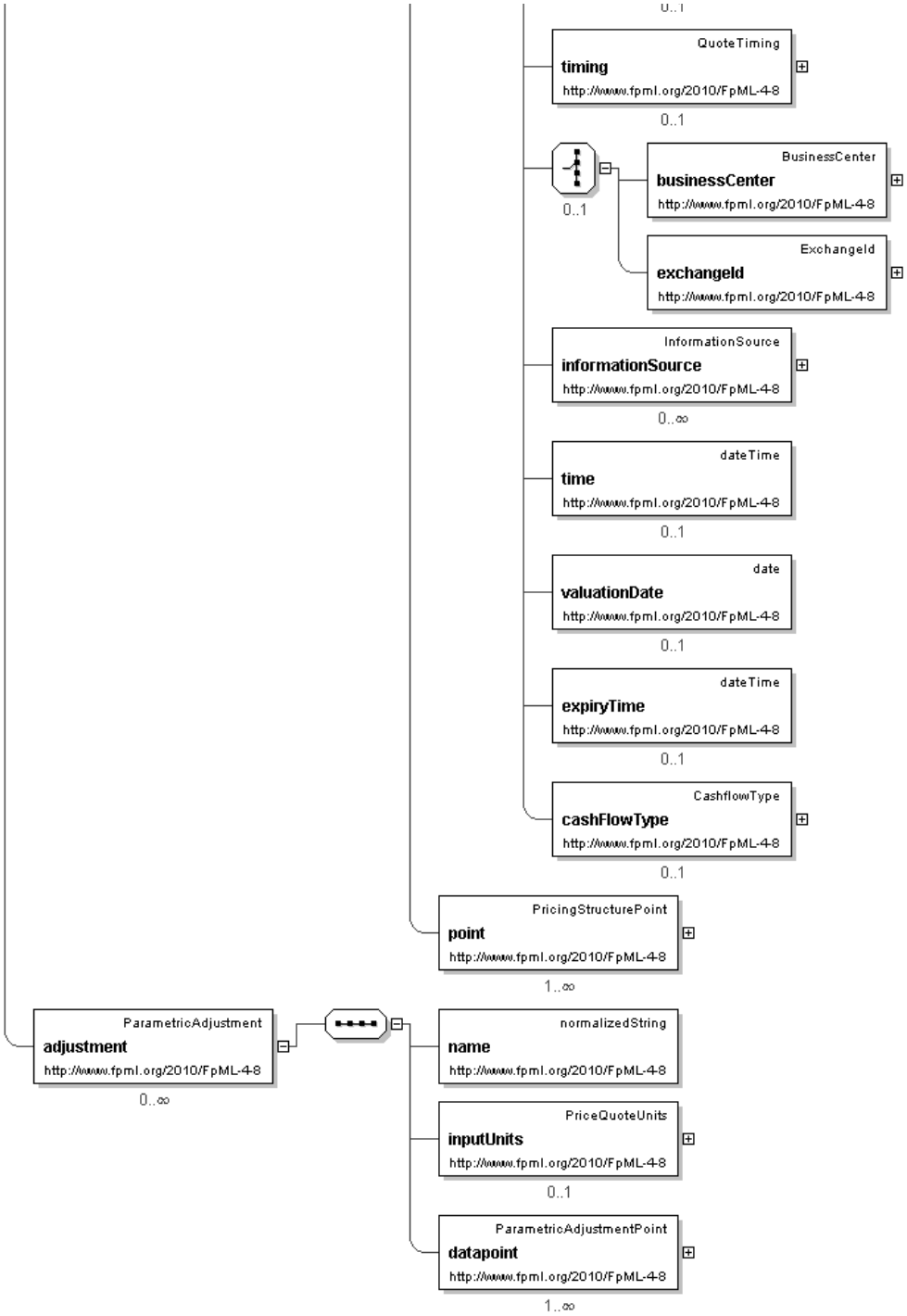
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructureValuation

Name	volatilityMatrixValuation
Type	VolatilityMatrix
Nillable	no
Abstract	no

Logical Diagram



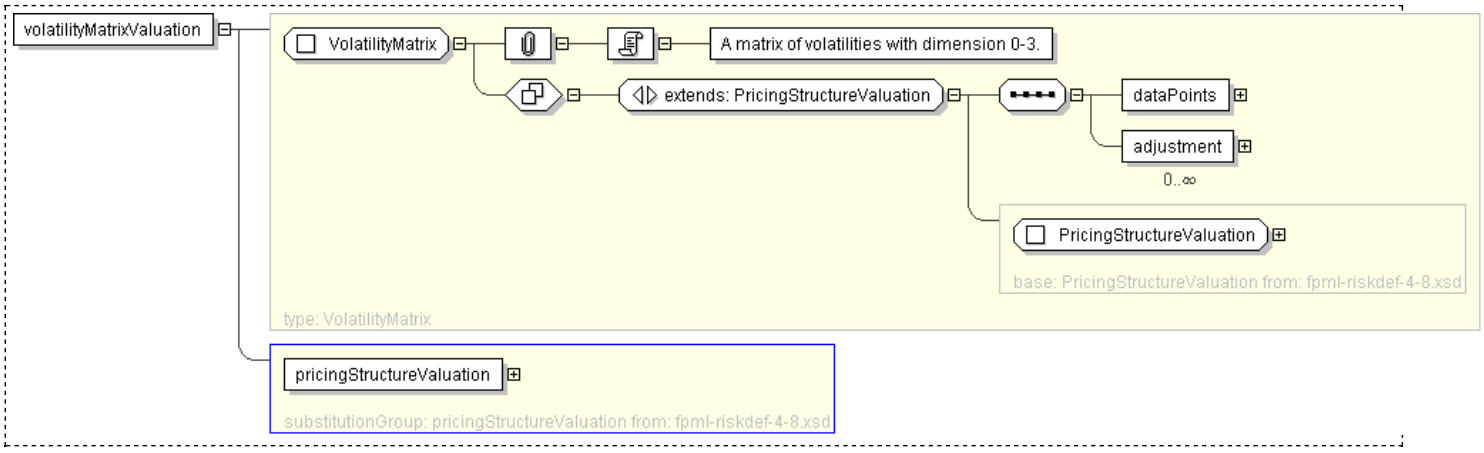


XML Instance Representation

```
<volatilityMatrixValuation
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a
   ValuationSet, this value is optional and is defaulted from the ValuationSet. If this value occurs in both places,
   the lower level value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation
   date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference
   between the baseDate and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the
```

```
    baseDate, but sometimes the pricing input may be \"rolled forward\", in which input data from one date is used to
    generate a curve for a later date.'

    <endDate> IdentifiedDate </endDate> [0..1]
    'The last date for which data is supplied in this pricing input.'

    <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
    'The date and time when the pricing input was generated.'

    <dataPoints> MultiDimensionalPricingData </dataPoints> [1]
    'The raw volatility matrix data, expressed as a multi-dimensional array.'

    <adjustment> ParametricAdjustment </adjustment> [0..*]
    'An adjustment factor, such as for vol smile/skew.'

</volatilityMatrixValuation>
```

Diagram



Schema Component Representation

```
<xsd:element name="volatilityMatrixValuation" type=" VolatilityMatrix " substitutionGroup="pricingStructureValuation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

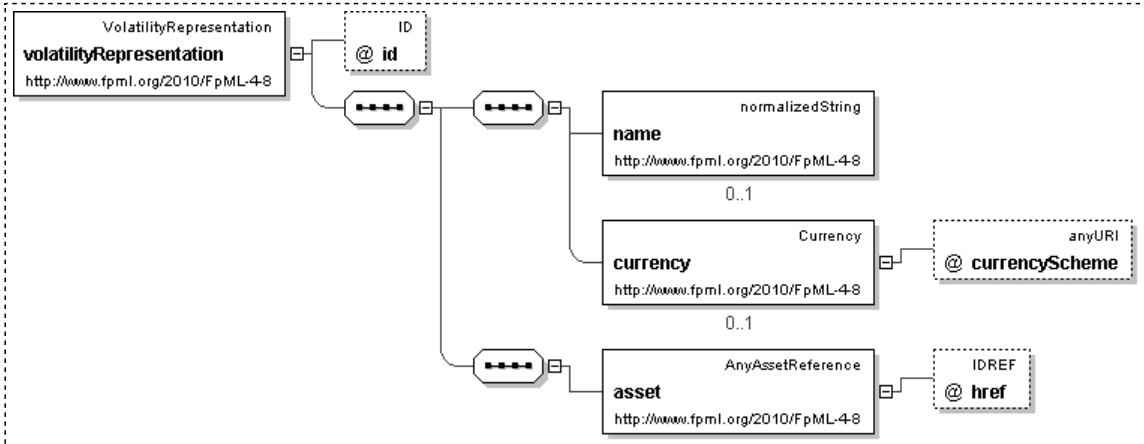
Element: volatilityRepresentation

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructure

Name	volatilityRepresentation
Type	VolatilityRepresentation
Nillable	no
Abstract	no

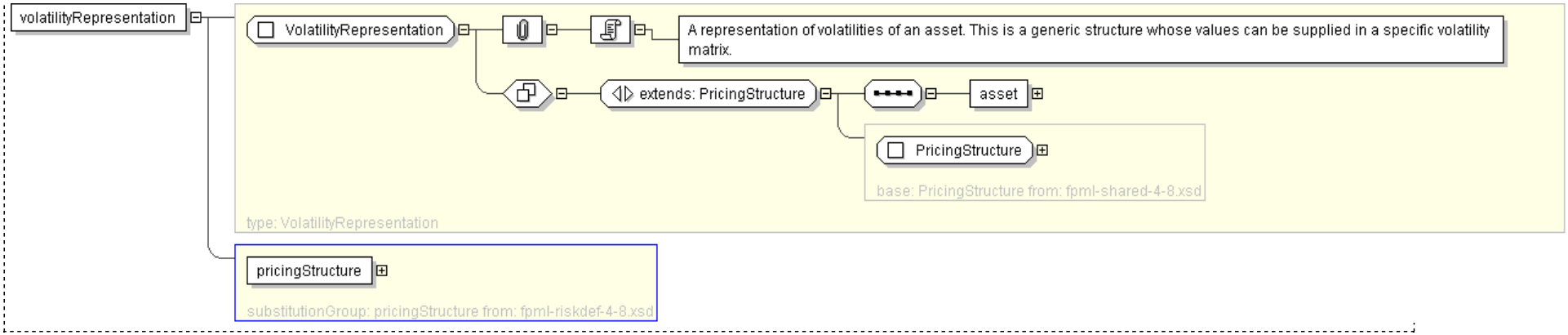
Logical Diagram



XML Instance Representation

```
<volatilityRepresentation
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates asset class).'

   <asset> AnyAssetReference </asset> [1]
   'A reference to the asset whose volatility is modeled.'

</volatilityRepresentation>
```

Diagram



Schema Component Representation

```
<xsd:element name="volatilityRepresentation" type=" VolatilityRepresentation " substitutionGroup="pricingStructure"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

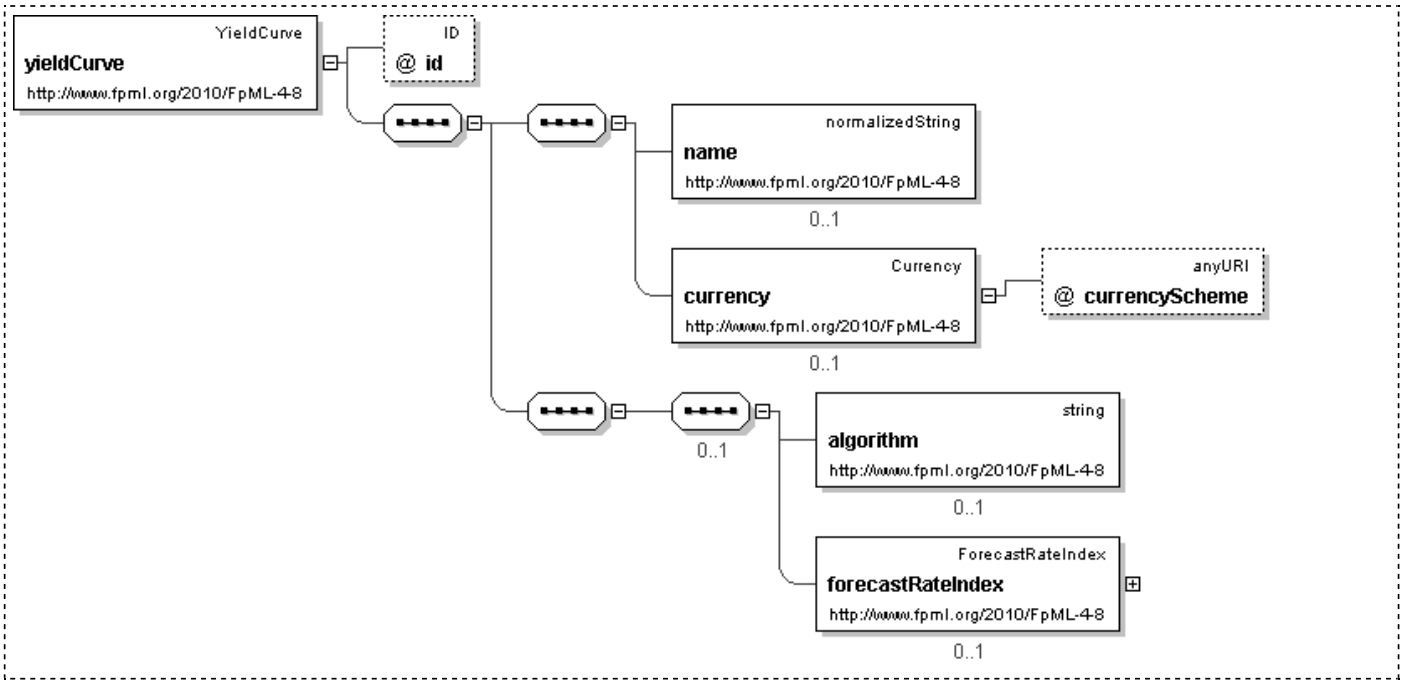
Element: yieldCurve

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructure

Name	yieldCurve
Type	YieldCurve
Nillable	no
Abstract	no

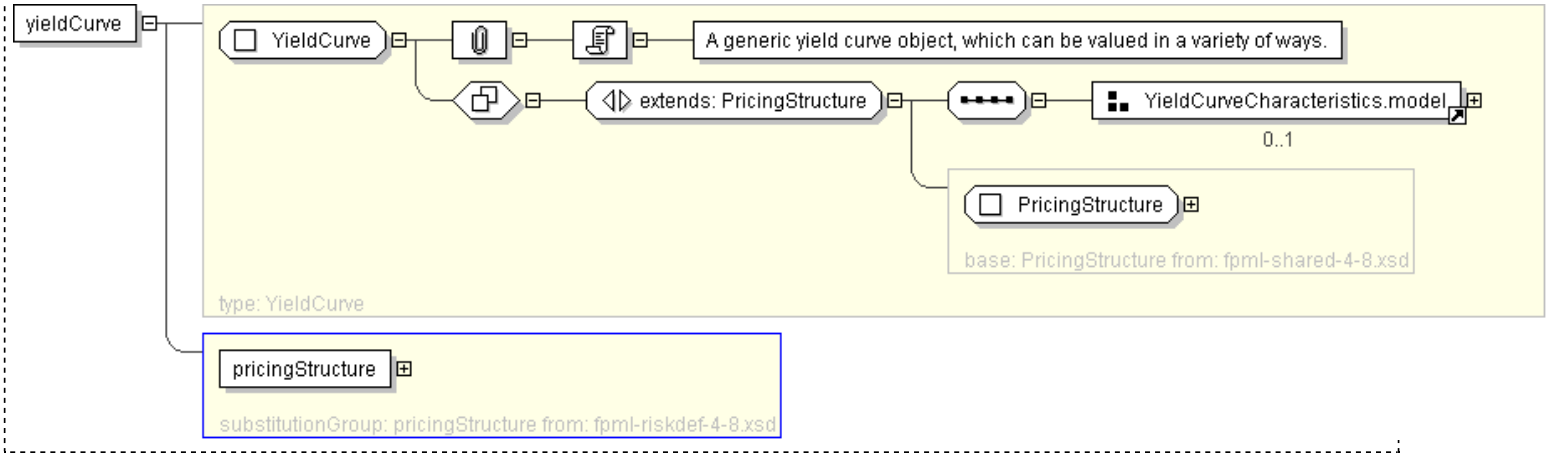
Logical Diagram



XML Instance Representation

```
<yieldCurve
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
    'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
    'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates
    asset class).'

   Start Group: YieldCurveCharacteristics.model [0..1]
      <algorithm> xsd:string </algorithm> [0..1]
      <forecastRateIndex> ForecastRateIndex </forecastRateIndex> [0..1]
   End Group: YieldCurveCharacteristics.model
</yieldCurve>
```

Diagram



Schema Component Representation

```
<xsd:element name="yieldCurve" type=" YieldCurve " substitutionGroup="pricingStructure"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

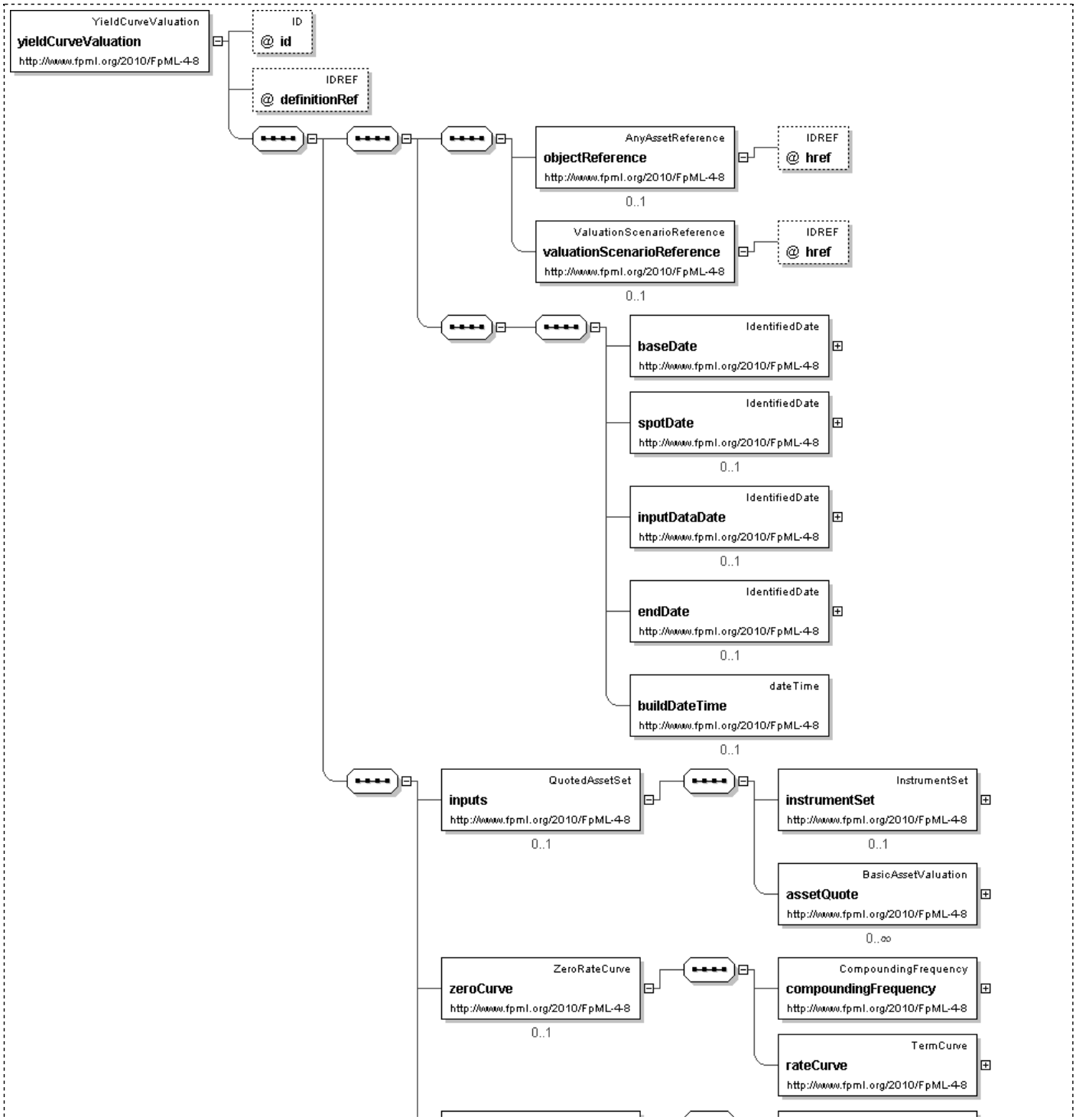
Element: yieldCurveValuation

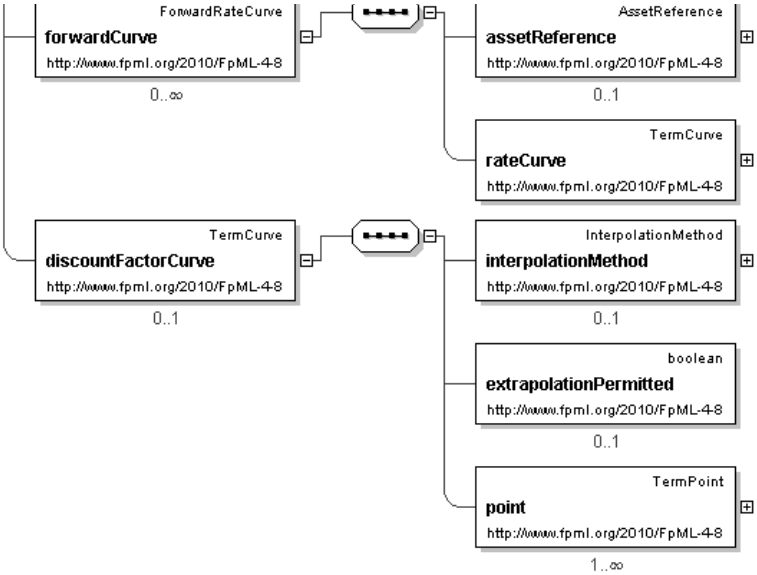
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - pricingStructureValuation

Name	yieldCurveValuation
Type	YieldCurveValuation
Nillable	no
Abstract	no

Logical Diagram

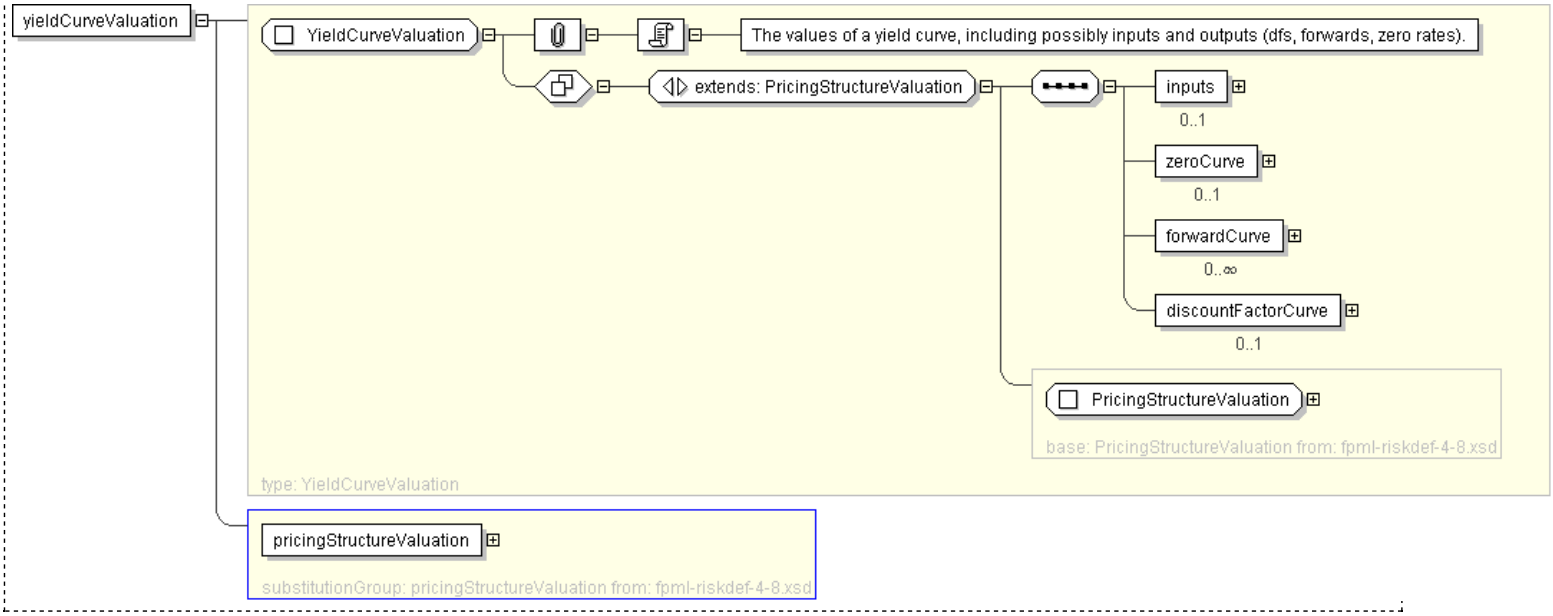




XML Instance Representation

```
<yieldCurveValuation
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a
   ValuationSet, this value is optional and is defaulted from the ValuationSet. If this value occurs in both places,
   the lower level value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation
   date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference
   between the baseDate and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the
   baseDate, but sometimes the pricing input may be \"rolled forward\", in which input data from one date is used to
   generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <inputs> QuotedAssetSet </inputs> [0..1]
   <zeroCurve> ZeroRateCurve </zeroCurve> [0..1]
   'A curve of zero rates.'

   <forwardCurve> ForwardRateCurve </forwardCurve> [0..*]
   'A curve of forward rates.'

   <discountFactorCurve> TermCurve </discountFactorCurve> [0..1]
   'A curve of discount factors.'

</yieldCurveValuation>
```

Diagram



Schema Component Representation

```
<xsd:element name="yieldCurveValuation" type=" YieldCurveValuation " substitutionGroup="pricingStructureValuation"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BidMidAsk.model

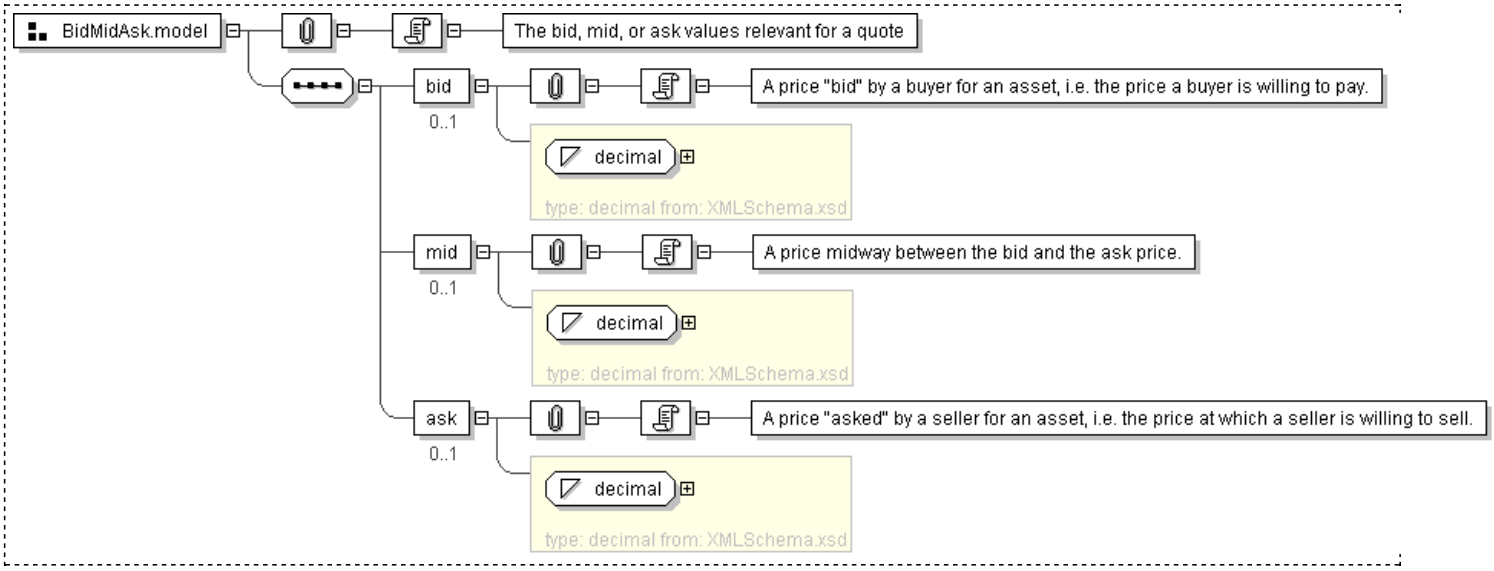
[Table of contents]

Name	BidMidAsk.model
Used by (from the same schema document)	Complex Type TermPoint
Documentation	The bid, mid, or ask values relevant for a quote

XML Instance Representation

```
<bid> xsd:decimal </bid> [0..1]
'A price \"bid\" by a buyer for an asset, i.e. the price a buyer is willing to pay.'

<mid> xsd:decimal </mid> [0..1]
'A price midway between the bid and the ask price.'

<ask> xsd:decimal </ask> [0..1]
'A price \"asked\" by a seller for an asset, i.e. the price at which a seller is willing to sell.'
```

Diagram



Schema Component Representation

```
<xsd:group name="BidMidAsk.model">
  <xsd:sequence>
    <xsd:element name="bid" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="mid" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="ask" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: CreditCurveCharacteristics.model

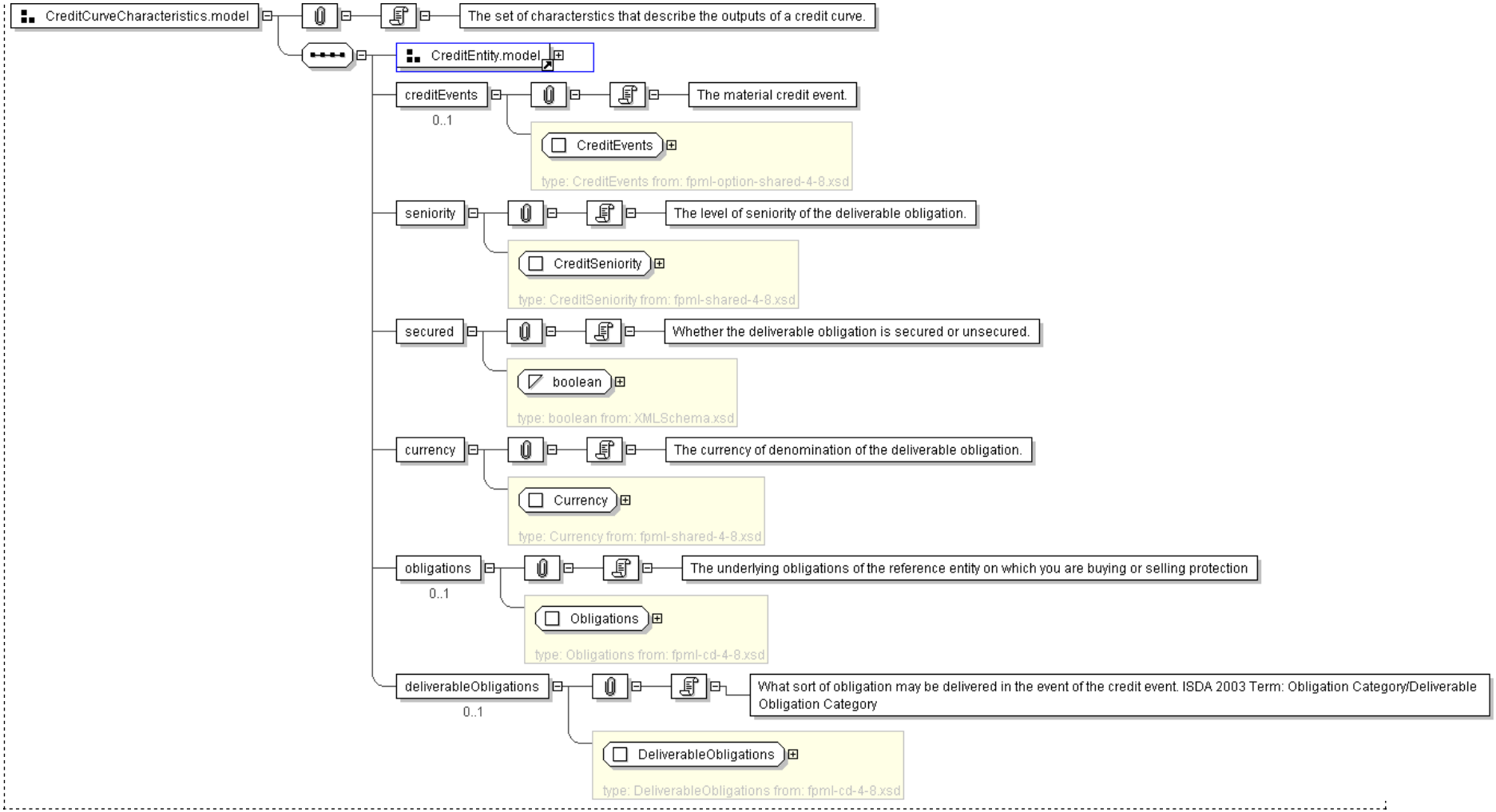
[Table of contents]

Name	CreditCurveCharacteristics.model
Used by (from the same schema document)	Complex Type CreditCurve
Documentation	The set of characterstics that describe the outputs of a credit curve.

XML Instance Representation

```
Start Choice [1]
   <referenceEntity> LegalEntity </referenceEntity> [1]
   'The entity for which this is defined.'

   <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
   'An XML reference a credit entity defined elsewhere in the document.'

End Choice
<creditEvents> CreditEvents </creditEvents> [0..1]
'The material credit event.'

<seniority> CreditSeniority </seniority> [1]
'The level of seniority of the deliverable obligation.'

<secured> xsd:boolean </secured> [1]
'Whether the deliverable obligation is secured or unsecured.'

<currency> Currency </currency> [1]
'The currency of denomination of the deliverable obligation.'

<obligations> Obligations </obligations> [0..1]
'The underlying obligations of the reference entity on which you are buying or selling protection'

<deliverableObligations> DeliverableObligations </deliverableObligations> [0..1]
'What sort of obligation may be delivered in the event of the credit event. ISDA 2003 Term: Obligation Category/Deliverable Obligation Category'
```

Diagram



Schema Component Representation

```
<xsd:group name="CreditCurveCharacteristics.model">
  <xsd:sequence>
    <xsd:group ref=" CreditEntity.model "/>
    <xsd:element name="creditEvents" type=" CreditEvents " minOccurs="0"/>
    <xsd:element name="seniority" type=" CreditSeniority "/>
    <xsd:element name="secured" type=" xsd:boolean "/>
    <xsd:element name="currency" type=" Currency "/>
    <xsd:element name="obligations" type=" Obligations " minOccurs="0"/>
    <xsd:element name="deliverableObligations" type=" DeliverableObligations " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: FxCurveCharacteristics.model

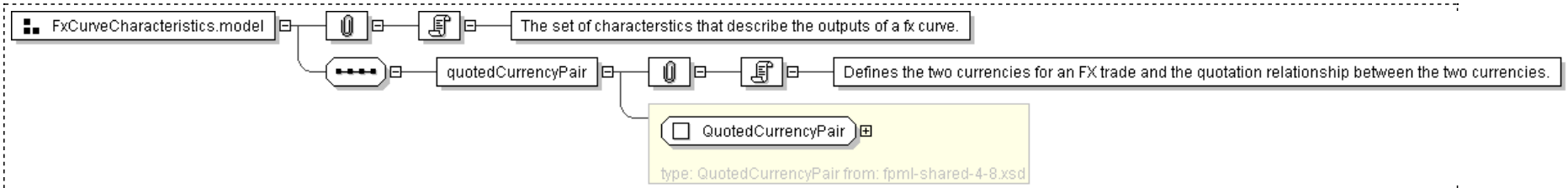
[Table of contents]

Name	FxCurveCharacteristics.model
Used by (from the same schema document)	Complex Type FxCurve
Documentation	The set of characterstics that describe the outputs of a fx curve.

XML Instance Representation

```
<quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'
```

Diagram



Schema Component Representation

```
<xsd:group name="FxCurveCharacteristics.model">
  <xsd:sequence>
    <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: RecoveryRate.model

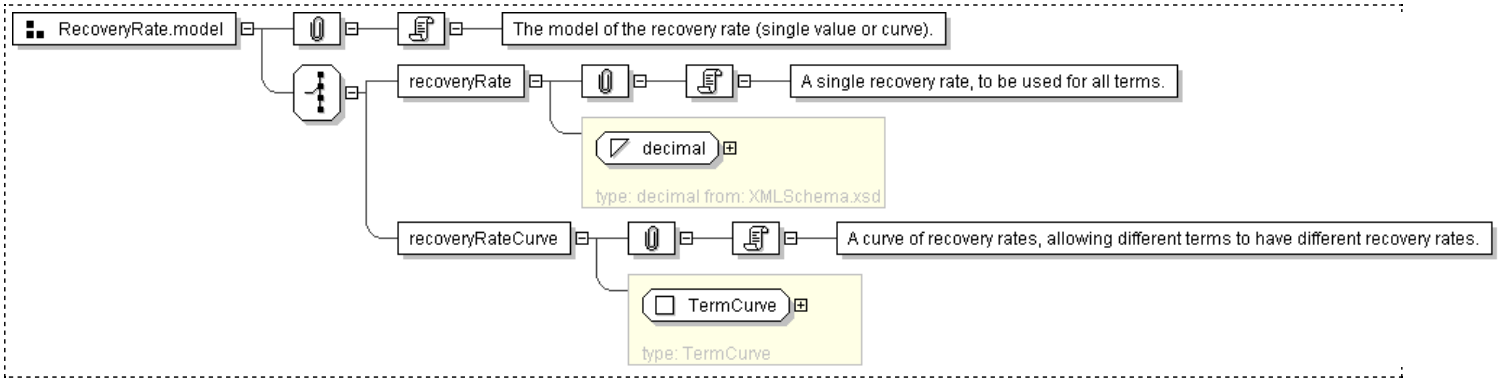
[Table of contents]

Name	RecoveryRate.model
Used by (from the same schema document)	Complex Type CreditCurveValuation
Documentation	The model of the recovery rate (single value or curve).

XML Instance Representation

```
Start Choice [1]
   <recoveryRate> xsd:decimal </recoveryRate> [1]
   'A single recovery rate, to be used for all terms.'

   <recoveryRateCurve> TermCurve </recoveryRateCurve> [1]
   'A curve of recovery rates, allowing different terms to have different recovery rates.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="RecoveryRate.model">
   <xsd:choice>
      <xsd:element name="recoveryRate" type=" xsd:decimal "/>
      <xsd:element name="recoveryRateCurve" type=" TermCurve "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: UnderlyingAssetOrReference.model

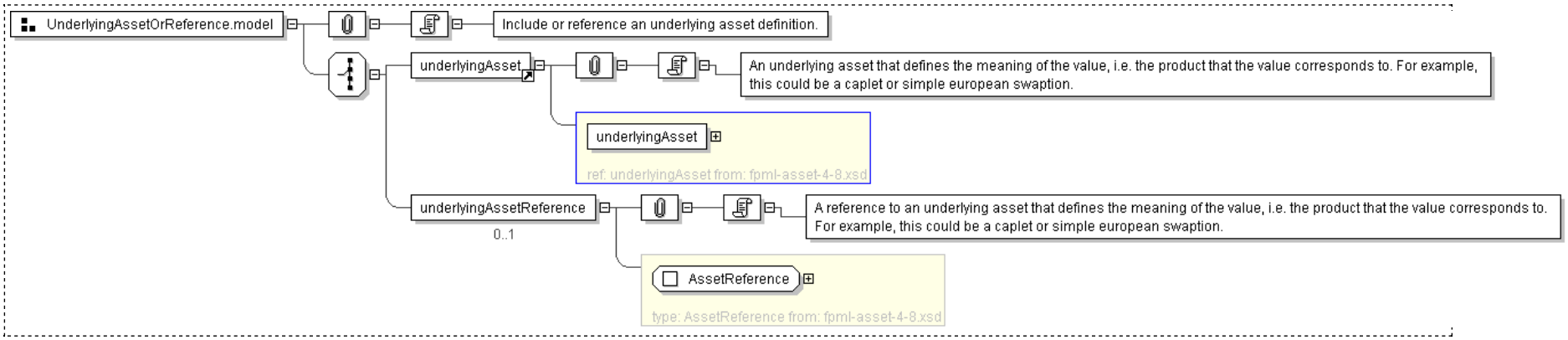
[Table of contents]

Name	UnderlyingAssetOrReference.model
Used by (from the same schema document)	Complex Type PricingStructurePoint
Documentation	Include or reference an underlying asset definition.

XML Instance Representation

```
Start Choice [1]
   <underlyingAsset> ... </underlyingAsset> [1]
   'An underlying asset that defines the meaning of the value, i.e. the product that the value corresponds to. For example, this could be a caplet or
   simple european swaption.'

   <underlyingAssetReference> AssetReference </underlyingAssetReference> [0..1]
   'A reference to an underlying asset that defines the meaning of the value, i.e. the product that the value corresponds to. For example, this could be a
   caplet or simple european swaption.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="UnderlyingAssetOrReference.model">
  <xsd:choice>
    <xsd:element ref=" underlyingAsset "/>
    <xsd:element name="underlyingAssetReference" type=" AssetReference " minOccurs="0"/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: YieldCurveCharacteristics.model

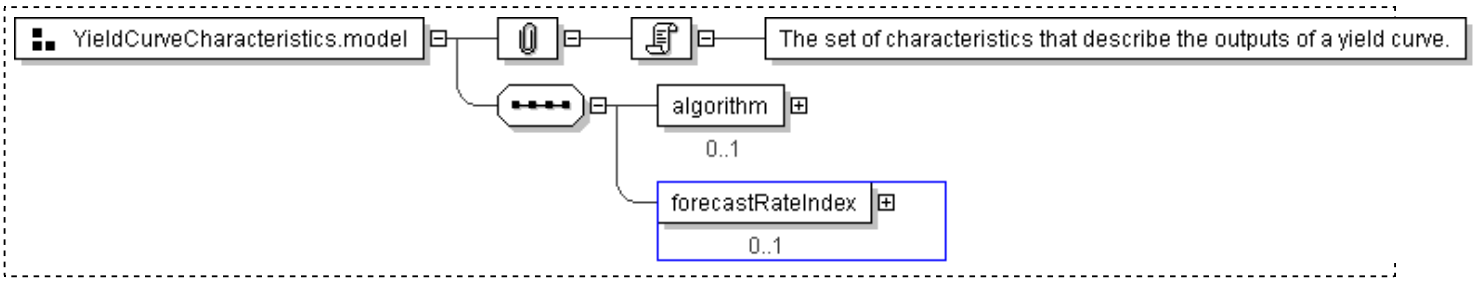
[Table of contents]

Name	YieldCurveCharacteristics.model
Used by (from the same schema document)	Complex Type YieldCurve
Documentation	The set of characteristics that describe the outputs of a yield curve.

XML Instance Representation

```
<algorithm> xsd:string </algorithm> [0..1]
<forecastRateIndex> ForecastRateIndex </forecastRateIndex> [0..1]
```

Diagram



Schema Component Representation

```
<xsd:group name="YieldCurveCharacteristics.model">
   <xsd:sequence>
      <xsd:element name="algorithm" type=" xsd:string " minOccurs="0"/>
      <xsd:element name="forecastRateIndex" type=" ForecastRateIndex " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CompoundingFrequency

[Table of contents]

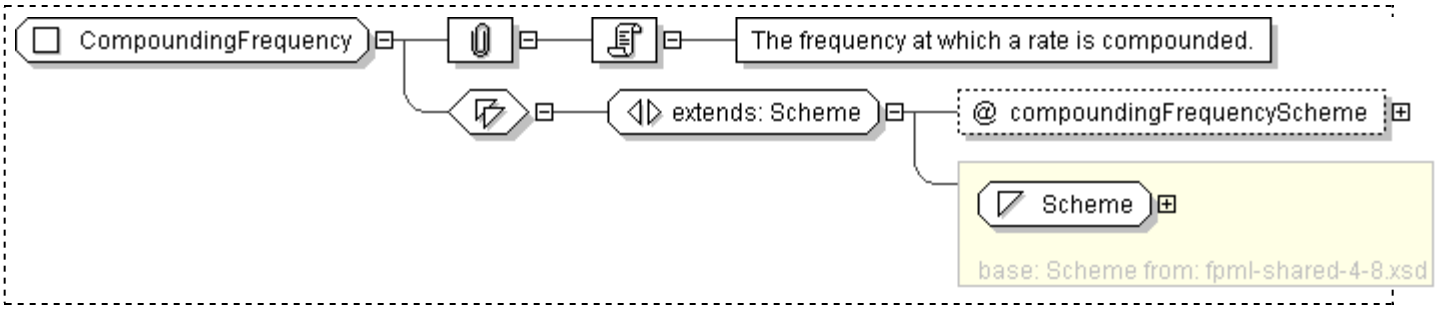
Super-types:	Scheme < CompoundingFrequency (by extension)
Sub-types:	None

Name	CompoundingFrequency
Used by (from the same schema document)	Complex Type ZeroRateCurve
Abstract	no
Documentation	The frequency at which a rate is compounded.

XML Instance Representation

```
<...
 compoundingFrequencyScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CompoundingFrequency">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="compoundingFrequencyScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/compounding-frequency"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditCurve

[Table of contents]

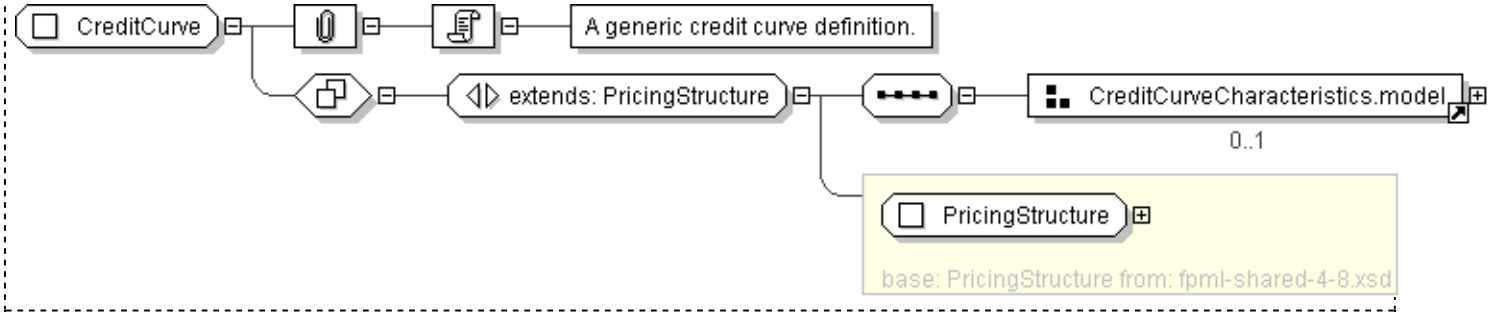
Super-types:	PricingStructure < CreditCurve (by extension)
Sub-types:	None

Name	CreditCurve
Used by (from the same schema document)	Element creditCurve
Abstract	no
Documentation	A generic credit curve definition.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the
   Interes Rates asset class).'

   Start Group: CreditCurveCharacteristics.model [0..1]
      Start Choice [1]
         <referenceEntity> LegalEntity </referenceEntity> [1]
         'The entity for which this is defined.'

         <creditEntityReference> LegalEntityReference </creditEntityReference> [1]
         'An XML reference a credit entity defined elsewhere in the document.'

      End Choice
      <creditEvents> CreditEvents </creditEvents> [0..1]
      'The material credit event.'

      <seniority> CreditSeniority </seniority> [1]
      'The level of seniority of the deliverable obligation.'

      <secured> xsd:boolean </secured> [1]
      'Whether the deliverable obligation is secured or unsecured.'

      <currency> Currency </currency> [1]
      'The currency of denomination of the deliverable obligation.'

      <obligations> Obligations </obligations> [0..1]
      'The underlying obligations of the reference entity on which you are buying or
      selling protection'

      <deliverableObligations> DeliverableObligations </deliverableObligations> [0..1]
      'What sort of obligation may be delivered in the event of the credit event. ISDA
      2003 Term: Obligation Category/Deliverable Obligation Category'

   End Group: CreditCurveCharacteristics.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditCurve">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructure ">
      <xsd:sequence>
        <xsd:group ref=" CreditCurveCharacteristics.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditCurveValuation

[Table of contents]

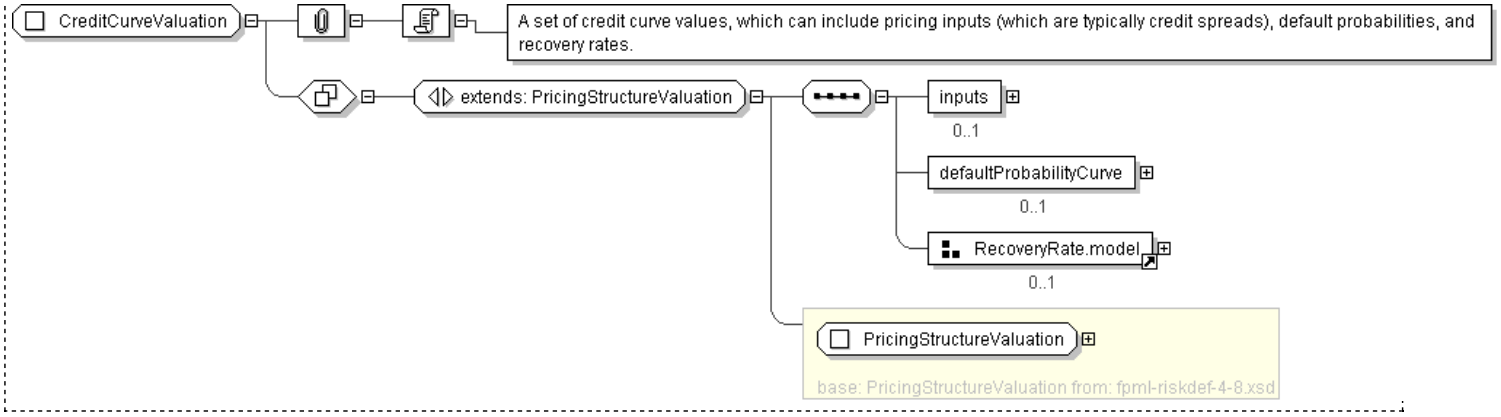
Super-types:	PricingStructureValuation < CreditCurveValuation (by extension)
Sub-types:	None

Name	CreditCurveValuation
Used by (from the same schema document)	Element creditCurveValuation
Abstract	no
Documentation	A set of credit curve values, which can include pricing inputs (which are typically credit spreads), default probabilities, and recovery rates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a
   ValuationSet, this value is optional and is defaulted from the ValuationSet. If this value occurs in both places,
   the lower level value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation
   date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference
   between the baseDate and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the
   baseDate, but sometimes the pricing input may be \"rolled forward\", in which input data from one date is used
   to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <inputs> QuotedAssetSet </inputs> [0..1]
   <defaultProbabilityCurve> DefaultProbabilityCurve </defaultProbabilityCurve> [0..1]
   'A curve of default probabilities.'

   Start Group: RecoveryRate.model [0..1]
'A recovery rate value or curve.'


     Start Choice [1]
       <recoveryRate> xsd:decimal </recoveryRate> [1]
       'A single recovery rate, to be used for all terms.'

       <recoveryRateCurve> TermCurve </recoveryRateCurve> [1]
       'A curve of recovery rates, allowing different terms to have different recovery rates.'

     End Choice
   End Group: RecoveryRate.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditCurveValuation">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructureValuation ">
      <xsd:sequence>
        <xsd:element name="inputs" type=" QuotedAssetSet " minOccurs="0"/>
        <xsd:element name="defaultProbabilityCurve" type=" DefaultProbabilityCurve " minOccurs="0"/>
        <xsd:group ref=" RecoveryRate.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DefaultProbabilityCurve

[Table of contents]

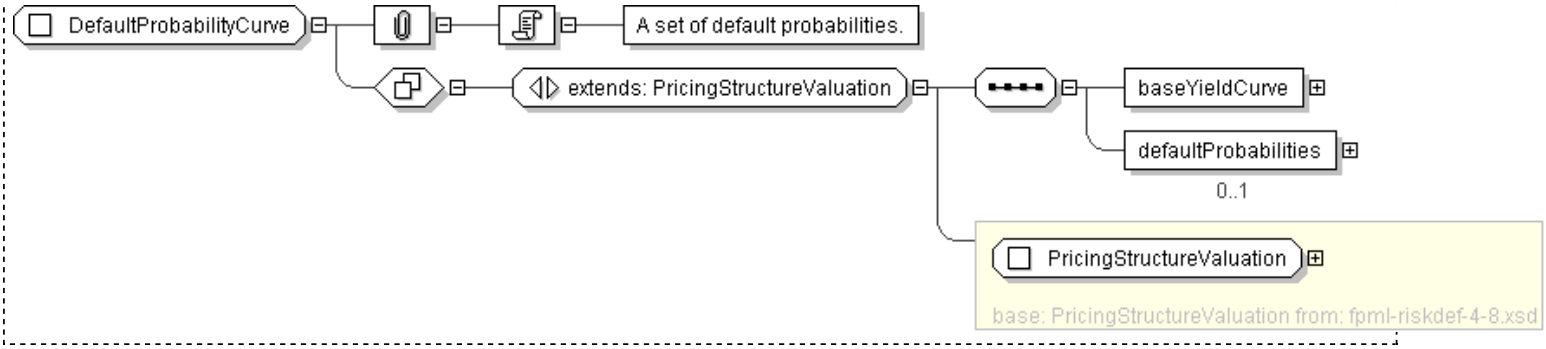
Super-types:	PricingStructureValuation < DefaultProbabilityCurve (by extension)
Sub-types:	None

Name	DefaultProbabilityCurve
Used by (from the same schema document)	Complex Type CreditCurveValuation
Abstract	no
Documentation	A set of default probabilities.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
 'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation
   occurs within a ValuationSet, this value is optional and is defaulted from the ValuationSet.
   If this value occurs in both places, the lower level value (i.e. the one here) overrides that
   in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align
   with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base
   date. The difference between the baseDate and the spotDate is termed the settlement lag, and
   is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often
   the same as the baseDate, but sometimes the pricing input may be \"rolled forward\", in which
   input data from one date is used to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <baseYieldCurve> PricingStructureReference </baseYieldCurve> [1]
   'A reference to the yield curve values used as a basis for this credit curve valuation.'

   <defaultProbabilities> TermCurve </defaultProbabilities> [0..1]
   'A collection of default probabilities.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DefaultProbabilityCurve">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructureValuation ">
      <xsd:sequence>
        <xsd:element name="baseYieldCurve" type=" PricingStructureReference "/>
        <xsd:element name="defaultProbabilities" type=" TermCurve " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ForwardRateCurve

[Table of contents]

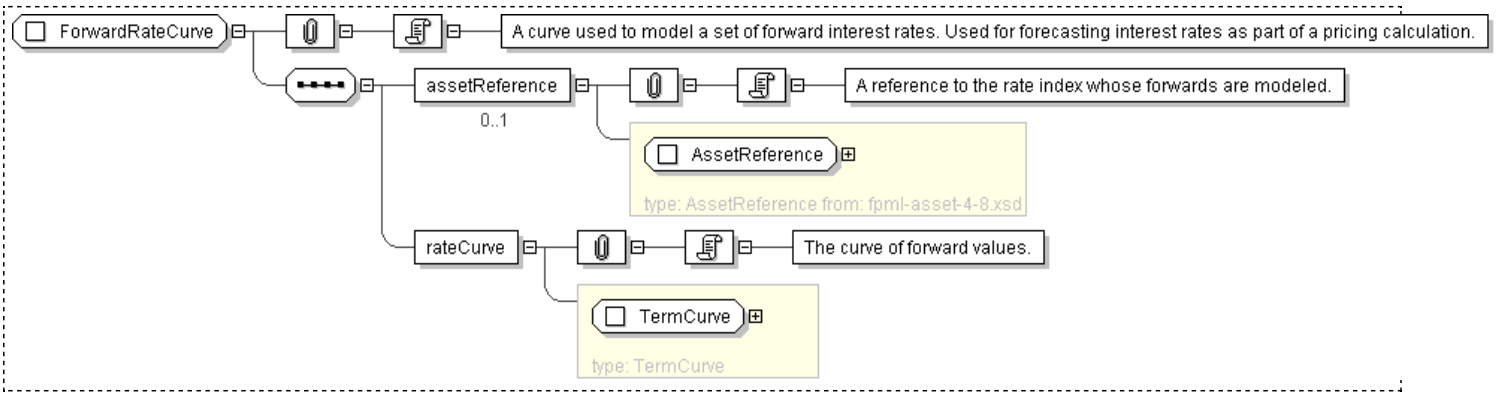
Super-types:	None
Sub-types:	None

Name	ForwardRateCurve
Used by (from the same schema document)	Complex Type YieldCurveValuation
Abstract	no
Documentation	A curve used to model a set of forward interest rates. Used for forecasting interest rates as part of a pricing calculation.

XML Instance Representation

```
<...>
  <assetReference> AssetReference </assetReference> [0..1]
  'A reference to the rate index whose forwards are modeled.'

  <rateCurve> TermCurve </rateCurve> [1]
  'The curve of forward values.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ForwardRateCurve">
  <xsd:sequence>
    <xsd:element name="assetReference" type=" AssetReference " minOccurs="0"/>
    <xsd:element name="rateCurve" type=" TermCurve "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxCurve

[Table of contents]

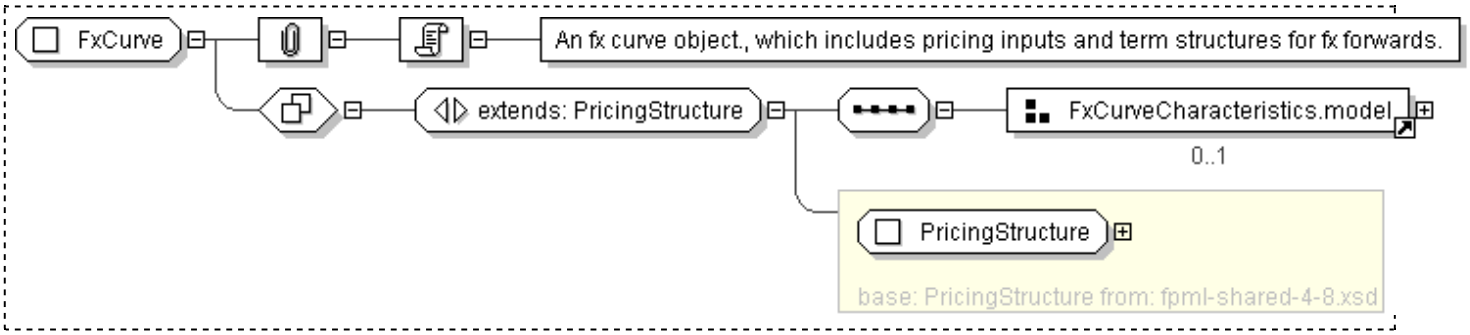
Super-types:	PricingStructure < FxCurve (by extension)
Sub-types:	None

Name	FxCurve
Used by (from the same schema document)	Element fxCurve
Abstract	no
Documentation	An fx curve object., which includes pricing inputs and term structures for fx forwards.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
    'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
    'The currency that the structure is expressed in (this is relevant mostly for the
    Interes Rates asset class).'

   Start Group: FxCurveCharacteristics.model [0..1]
      <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
       'Defines the two currencies for an FX trade and the quotation relationship
       between the two currencies.'

   End Group: FxCurveCharacteristics.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxCurve">
   <xsd:complexContent>
      <xsd:extension base=" PricingStructure ">
         <xsd:sequence>
            <xsd:group ref=" FxCurveCharacteristics.model " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxCurveValuation

[Table of contents]

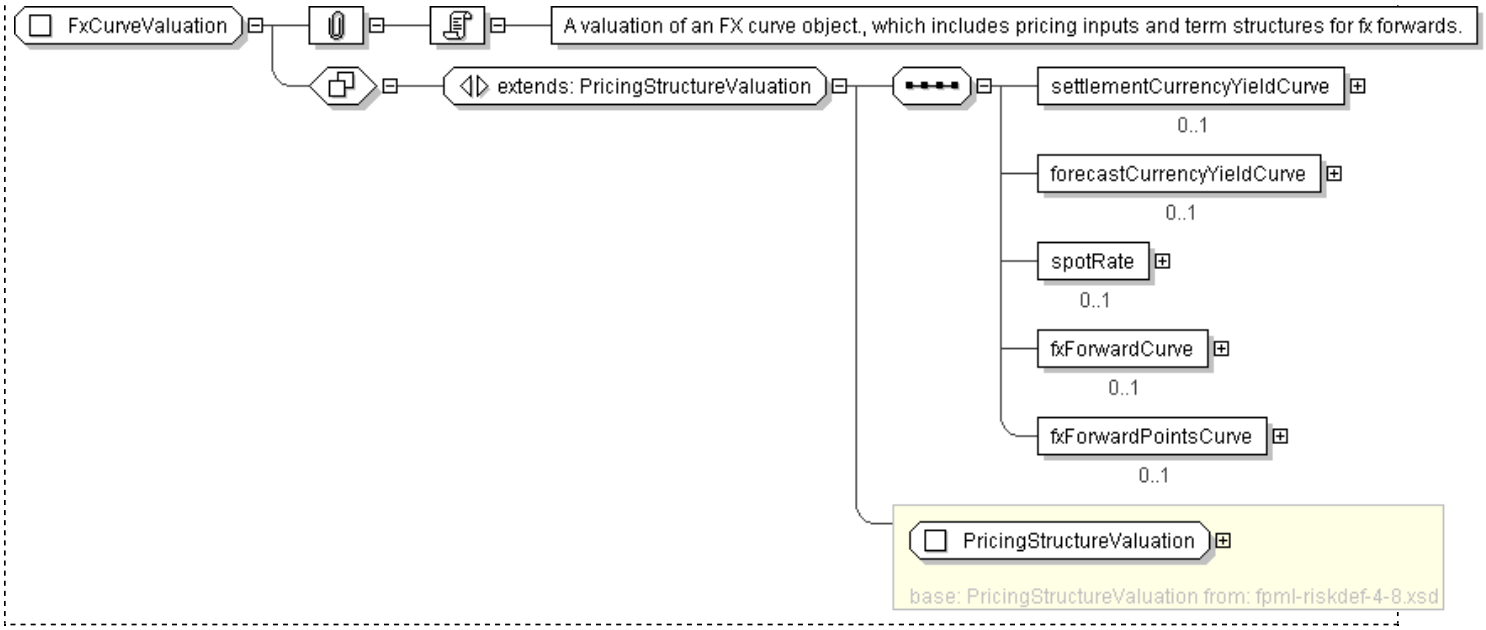
Super-types:	PricingStructureValuation < FxCurveValuation (by extension)
Sub-types:	None

Name	FxCurveValuation
Used by (from the same schema document)	Element fxCurveValuation
Abstract	no
Documentation	A valuation of an FX curve object., which includes pricing inputs and term structures for fx forwards.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
 'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs
   within a ValuationSet, this value is optional and is defaulted from the ValuationSet. If this
   value occurs in both places, the lower level value (i.e. the one here) overrides that in the
   higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align
   with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base
   date. The difference between the baseDate and the spotDate is termed the settlement lag, and is
   sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the
   same as the baseDate, but sometimes the pricing input may be \"rolled forward\", in which input
   data from one date is used to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <settlementCurrencyYieldCurve> PricingStructureReference </settlementCurrencyYieldCurve> [0..1]
   <forecastCurrencyYieldCurve> PricingStructureReference </forecastCurrencyYieldCurve> [0..1]
   <spotRate> FxRateSet </spotRate> [0..1]
   <fxForwardCurve> TermCurve </fxForwardCurve> [0..1]
   'A curve of fx forward rates.'

   <fxForwardPointsCurve> TermCurve </fxForwardPointsCurve> [0..1]
   'A curve of fx forward point spreads.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxCurveValuation">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructureValuation ">
      <xsd:sequence>
        <xsd:element name="settlementCurrencyYieldCurve" type=" PricingStructureReference "
        minOccurs="0"/>
        <xsd:element name="forecastCurrencyYieldCurve" type=" PricingStructureReference "
        minOccurs="0"/>
        <xsd:element name="spotRate" type=" FxRateSet " minOccurs="0"/>
        <xsd:element name="fxForwardCurve" type=" TermCurve " minOccurs="0"/>
        <xsd:element name="fxForwardPointsCurve" type=" TermCurve " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxRateSet

[Table of contents]

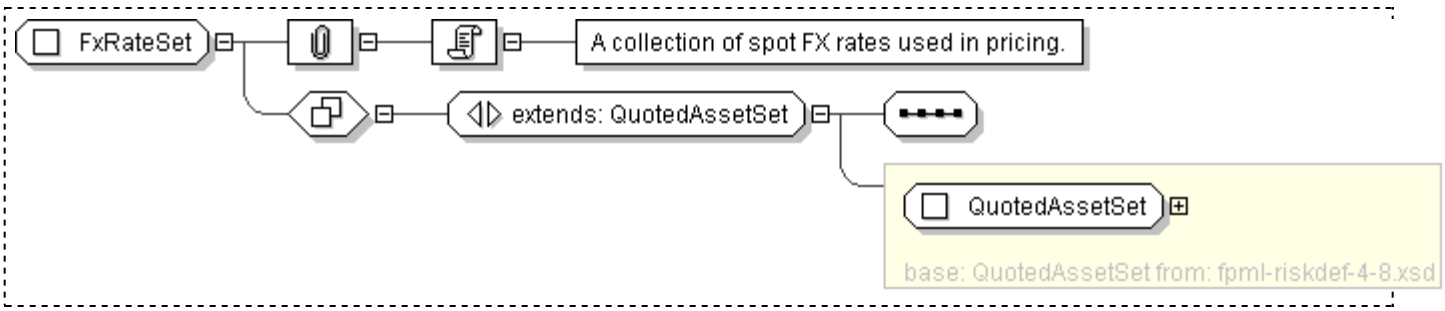
Super-types:	QuotedAssetSet < FxRateSet (by extension)
Sub-types:	None

Name	FxRateSet
Used by (from the same schema document)	Complex Type FxCurveValuation
Abstract	no
Documentation	A collection of spot FX rates used in pricing.

XML Instance Representation

```
<...>
   <instrumentSet> InstrumentSet </instrumentSet> [0..1]
   'A collection of instruments used as a basis for quotation.'

   <assetQuote> BasicAssetValuation </assetQuote> [0..*]
   'A collection of valuations (quotes) for the assets needed in the set. Normally
   these quotes will be for the underlying assets listed above, but they don\'t
   necesarily have to be.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxRateSet">
   <xsd:complexContent>
      <xsd:extension base=" QuotedAssetSet ">
         <xsd:sequence/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MultiDimensionalPricingData

[Table of contents]

Super-types:	None
Sub-types:	None

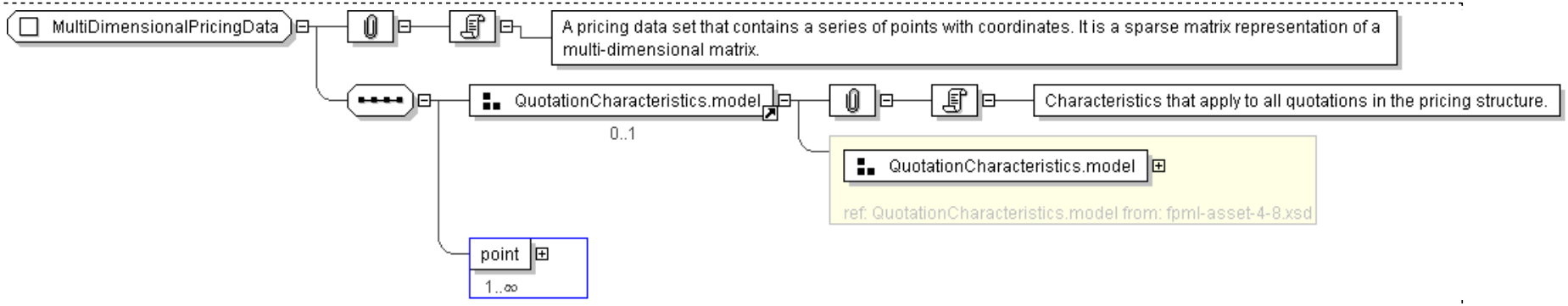
Name	MultiDimensionalPricingData
Used by (from the same schema document)	Complex Type VolatilityMatrix
Abstract	no
Documentation	A pricing data set that contains a series of points with coordinates. It is a sparse matrix representation of a multi-dimensional matrix.

XML Instance Representation

```
<...>
   Start Group: QuotationCharacteristics.model [0..1]
'Characteristics that apply to all quotations in the pricing structure.'


      <measureType> AssetMeasureType </measureType> [0..1]
      'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

      <quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
      'The optional units that the measure is expressed in. If not supplied, this is assumed to be a price/value in currency
      units.'

      <side> QuotationSideEnum </side> [0..1]
      'The side (bid/mid/ask) of the measure.'

      <currency> Currency </currency> [0..1]
      'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the reportingCurrency
      in the valuationScenarioDefinition.'

      <timing> QuoteTiming </timing> [0..1]
      'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs also to be
      supplied.'

      Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


         Start Choice [1]
            <businessCenter> BusinessCenter </businessCenter> [1]
            'A city or other business center.'

            <exchangeId> ExchangeId </exchangeId> [1]
            'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

         End Choice
      End Group: QuoteLocation.model
      <informationSource> InformationSource </informationSource> [0..*]
      'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

      <time> xsd:dateTime </time> [0..1]
```

```
      'When the quote was observed or derived.'

      <valuationDate> xsd:date </valuationDate> [0..1]
      'When the quote was computed.'

      <expiryTime> xsd:dateTime </expiryTime> [0..1]
      'When does the quote cease to be valid.'

      <cashFlowType> CashflowType </cashFlowType> [0..1]
      'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement Fee, Brokerage
      Fee, etc.'

   End Group: QuotationCharacteristics.model
   <point> PricingStructurePoint </point> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MultiDimensionalPricingData">
   <xsd:sequence>
      <xsd:group ref=" QuotationCharacteristics.model " minOccurs="0"/>
      <xsd:element name="point" type=" PricingStructurePoint " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ParametricAdjustment

[Table of contents]

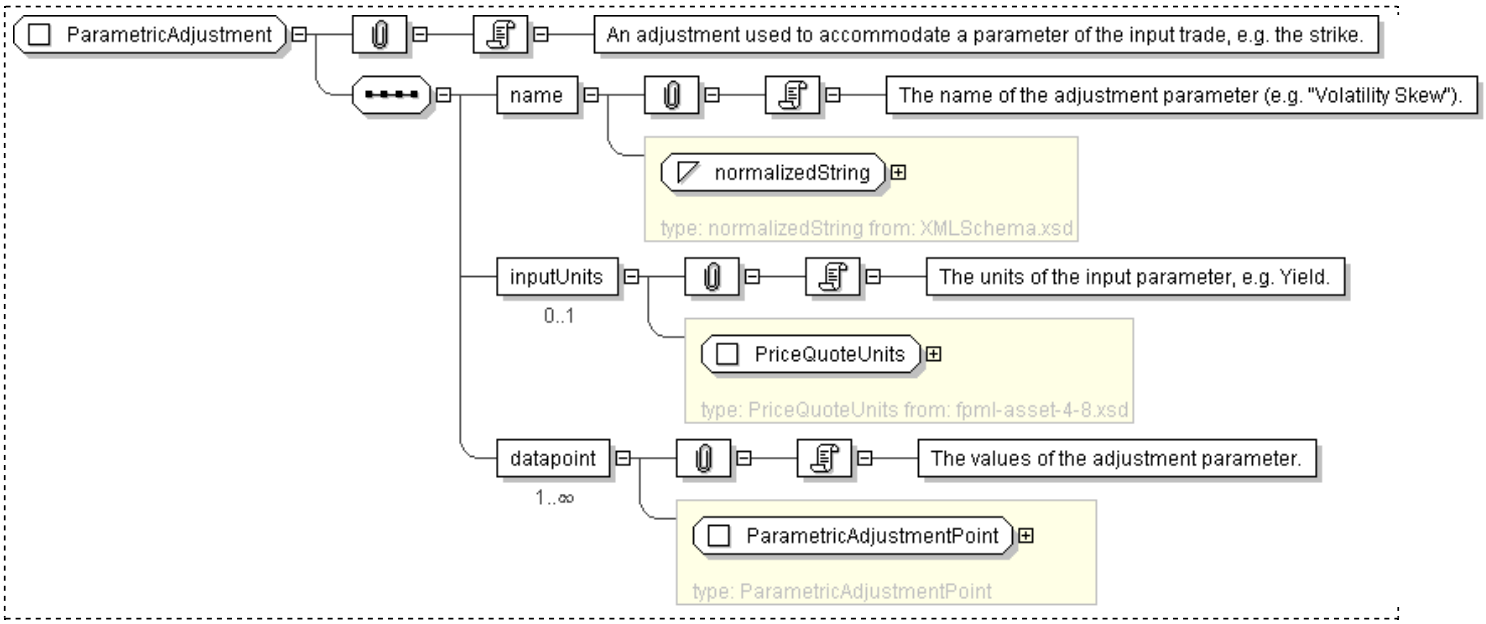
Super-types:	None
Sub-types:	None

Name	ParametricAdjustment
Used by (from the same schema document)	Complex Type VolatilityMatrix
Abstract	no
Documentation	An adjustment used to accommodate a parameter of the input trade, e.g. the strike.

XML Instance Representation

```
<...>
   <name> xsd:normalizedString </name> [1]
   'The name of the adjustment parameter (e.g. \"Volatility Skew\").'

   <inputUnits> PriceQuoteUnits </inputUnits> [0..1]
   'The units of the input parameter, e.g. Yield.'

   <datapoint> ParametricAdjustmentPoint </datapoint> [1..*]
   'The values of the adjustment parameter.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ParametricAdjustment">
   <xsd:sequence>
      <xsd:element name="name" type=" xsd:normalizedString "/>
      <xsd:element name="inputUnits" type=" PriceQuoteUnits " minOccurs="0"/>
      <xsd:element name="datapoint" type=" ParametricAdjustmentPoint " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ParametricAdjustmentPoint

[Table of contents]

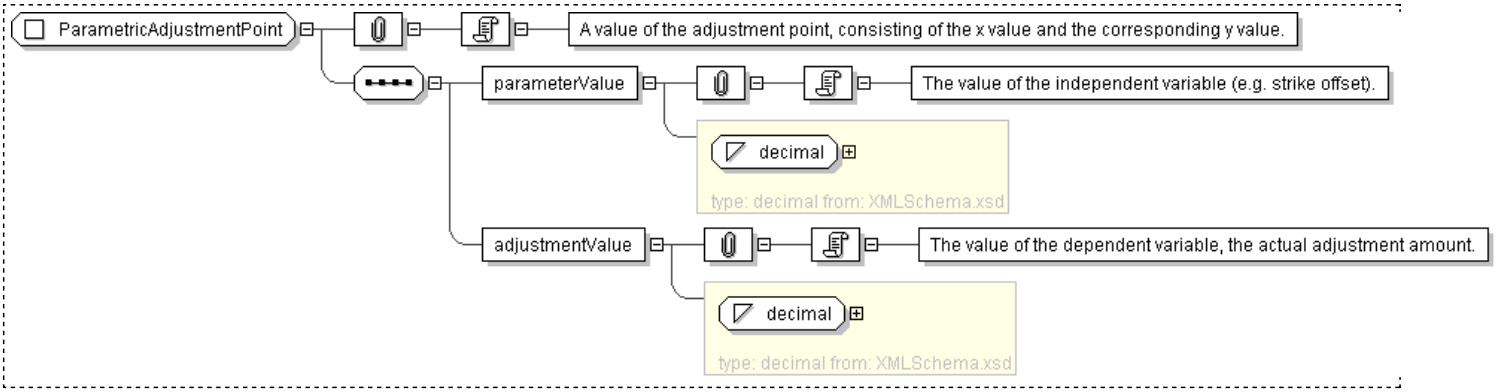
Super-types:	None
Sub-types:	None

Name	ParametricAdjustmentPoint
Used by (from the same schema document)	Complex Type ParametricAdjustment
Abstract	no
Documentation	A value of the adjustment point, consisting of the x value and the corresponding y value.

XML Instance Representation

```
<...>
  <parameterValue> xsd:decimal </parameterValue> [1]
  'The value of the independent variable (e.g. strike offset).'

  <adjustmentValue> xsd:decimal </adjustmentValue> [1]
  'The value of the dependent variable, the actual adjustment amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ParametricAdjustmentPoint">
  <xsd:sequence>
    <xsd:element name="parameterValue" type=" xsd:decimal "/>
    <xsd:element name="adjustmentValue" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingStructurePoint

[Table of contents]

Super-types:	None
Sub-types:	None

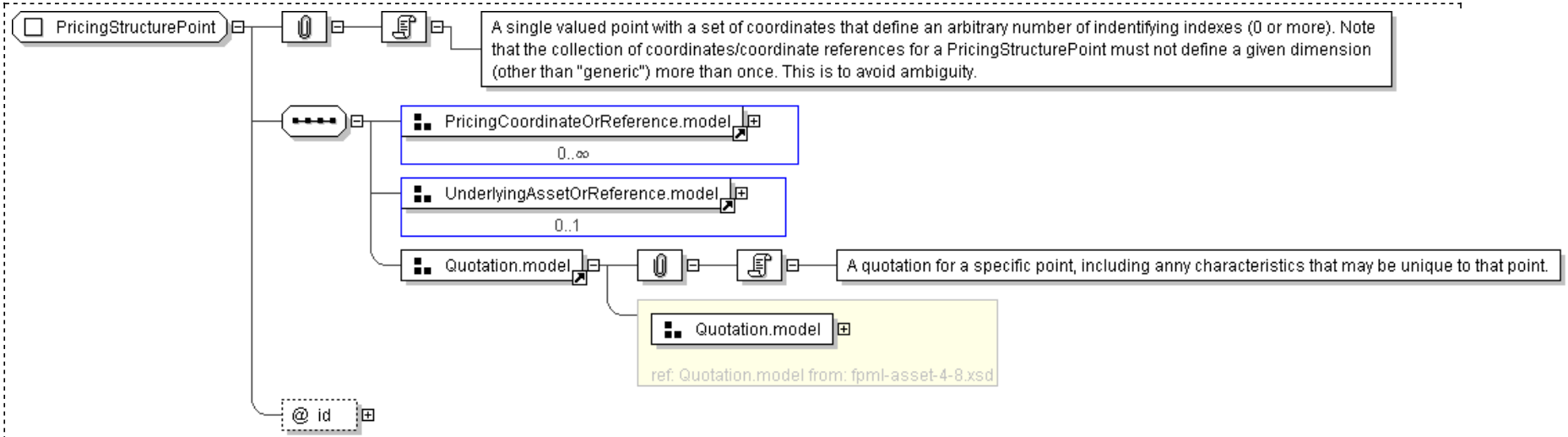
Name	PricingStructurePoint
Used by (from the same schema document)	Complex Type MultiDimensionalPricingData
Abstract	no
Documentation	A single valued point with a set of coordinates that define an arbitrary number of indentifying indexes (0 or more). Note that the collection of coordinates/coordinate references for a PricingStructurePoint must not define a given dimension (other than "generic") more than once. This is to avoid ambiguity.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Group: PricingCoordinateOrReference.model [0..*]
      Start Choice [1]
         <coordinate> PricingDataPointCoordinate </coordinate> [1]
         'An explicit, filled in data point coordinate. This might specify expiration, strike, etc.'

         <coordinateReference> PricingDataPointCoordinateReference </coordinateReference> [1]
         'A reference to a pricing data point coordinate within this document.'

      End Choice
   End Group: PricingCoordinateOrReference.model
   Start Group: UnderlyingAssetOrReference.model [0..1]
      Start Choice [1]
         <underlyingAsset> ... </underlyingAsset> [1]
         'An underlying asset that defines the meaning of the value, i.e. the product that the value corresponds to. For
         example, this could be a caplet or simple european swaption.'

         <underlyingAssetReference> AssetReference </underlyingAssetReference> [0..1]
         'A reference to an underlying asset that defines the meaning of the value, i.e. the product that the value corresponds
         to. For example, this could be a caplet or simple european swaption.'

      End Choice
   End Group: UnderlyingAssetOrReference.model
   <value> xsd:decimal </value> [0..1]
   'The value of the the quotation.'

   <measureType> AssetMeasureType </measureType> [0..1]
   'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

   <quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
   'The optional units that the measure is expressed in. If not supplied, this is assumed to be a price/value in currency
   units.'

   <side> QuotationSideEnum </side> [0..1]
   'The side (bid/mid/ask) of the measure.'

   <currency> Currency </currency> [0..1]
   'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the reportingCurrency in
   the valuationScenarioDefinition.'
```

```
   <timing> QuoteTiming </timing> [0..1]
   'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs also to be supplied.'

   Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


      Start Choice [1]
         <businessCenter> BusinessCenter </businessCenter> [1]
         'A city or other business center.'

         <exchangeId> ExchangeId </exchangeId> [1]
         'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

      End Choice
   End Group: QuoteLocation.model
   <informationSource> InformationSource </informationSource> [0..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

   <time> xsd:dateTime </time> [0..1]
   'When the quote was observed or derived.'

   <valuationDate> xsd:date </valuationDate> [0..1]
   'When the quote was computed.'

   <expiryTime> xsd:dateTime </expiryTime> [0..1]
   'When does the quote cease to be valid.'

   <cashFlowType> CashflowType </cashFlowType> [0..1]
   'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement Fee, Brokerage Fee,
   etc.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingStructurePoint">
   <xsd:sequence>
      <xsd:group ref=" PricingCoordinateOrReference.model " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:group ref=" UnderlyingAssetOrReference.model " minOccurs="0"/>
      <xsd:group ref=" Quotation.model "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TermCurve

[Table of contents]

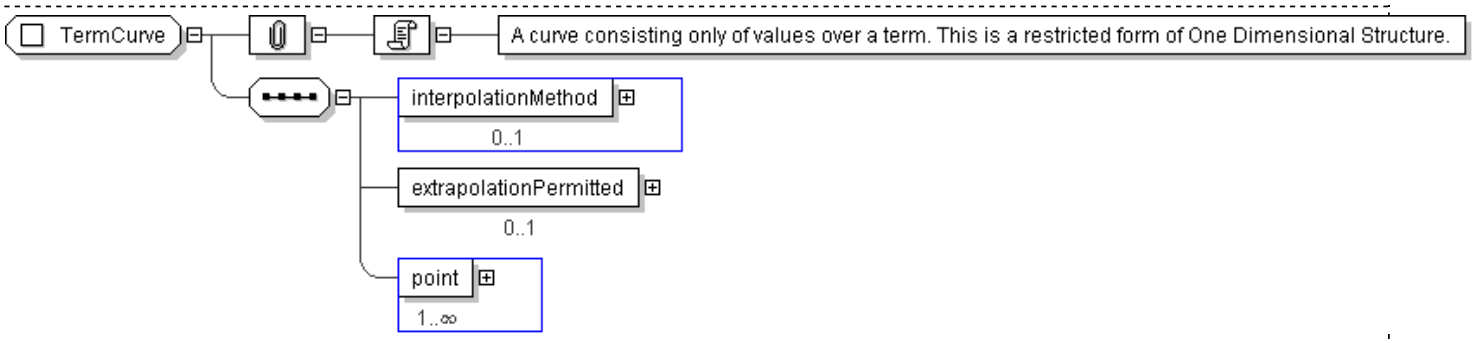
Super-types:	None
Sub-types:	None

Name	TermCurve
Used by (from the same schema document)	Complex Type DefaultProbabilityCurve , Complex Type ForwardRateCurve , Complex Type FxCurveValuation , Complex Type FxCurveValuation , Complex Type YieldCurveValuation , Complex Type ZeroRateCurve , Model Group RecoveryRate.model
Abstract	no
Documentation	A curve consisting only of values over a term. This is a restricted form of One Dimensional Structure.

XML Instance Representation

```
<...>
   <interpolationMethod> InterpolationMethod </interpolationMethod> [0..1]
   <extrapolationPermitted> xsd:boolean </extrapolationPermitted> [0..1]
   <point> TermPoint </point> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TermCurve">
   <xsd:sequence>
      <xsd:element name="interpolationMethod" type=" InterpolationMethod " minOccurs="0"/>
      <xsd:element name="extrapolationPermitted" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="point" type=" TermPoint " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TermPoint

[Table of contents]

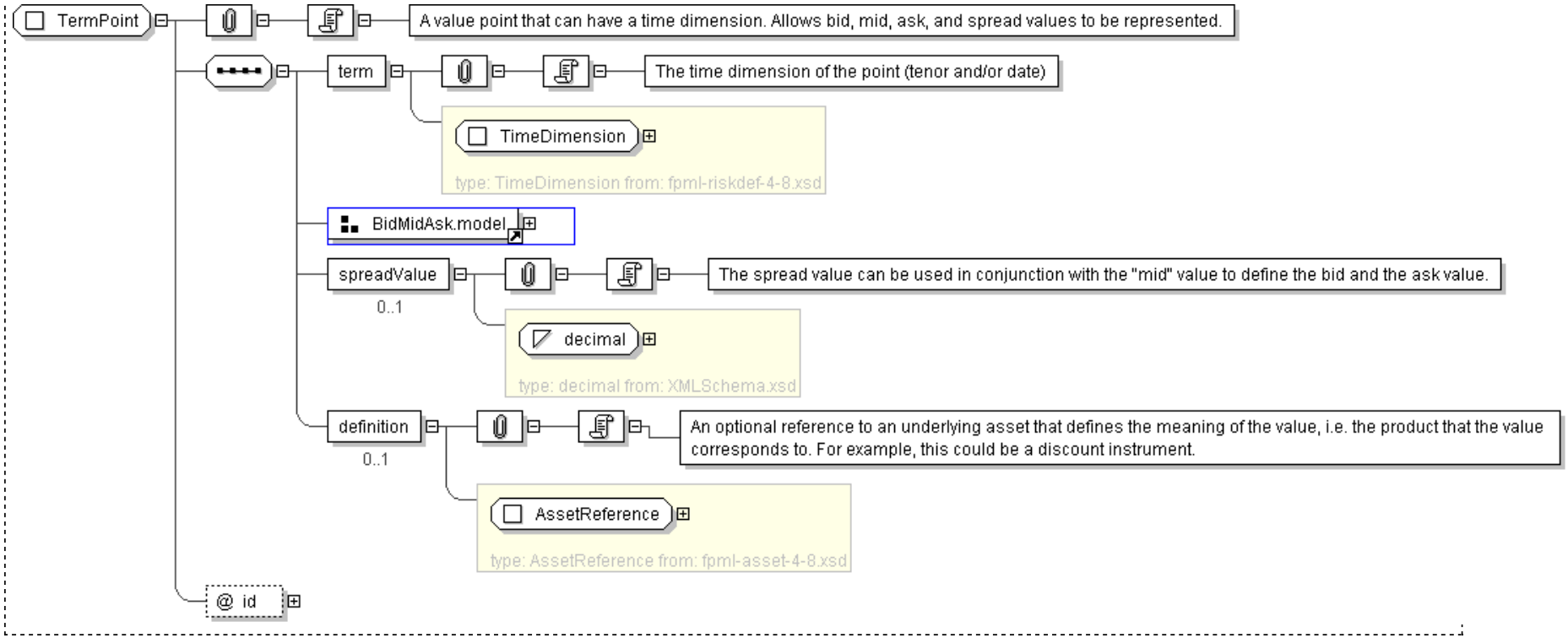
Super-types:	None
Sub-types:	None

Name	TermPoint
Used by (from the same schema document)	Complex Type TermCurve
Abstract	no
Documentation	A value point that can have a time dimension. Allows bid, mid, ask, and spread values to be represented.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <term> TimeDimension </term> [1]
    'The time dimension of the point (tenor and/or date)'

   <bid> xsd:decimal </bid> [0..1]
    'A price \"bid\" by a buyer for an asset, i.e. the price a buyer is willing to pay.'

   <mid> xsd:decimal </mid> [0..1]
    'A price midway between the bid and the ask price.'

   <ask> xsd:decimal </ask> [0..1]
    'A price \"asked\" by a seller for an asset, i.e. the price at which a seller is willing to sell.'

   <spreadValue> xsd:decimal </spreadValue> [0..1]
    'The spread value can be used in conjunction with the \"mid\" value to define the bid and the ask value.'

   <definition> AssetReference </definition> [0..1]
    'An optional reference to an underlying asset that defines the meaning of the value, i.e. the product that the value
    corresponds to. For example, this could be a discount instrument.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TermPoint">
  <xsd:sequence>
    <xsd:element name="term" type=" TimeDimension "/>
    <xsd:group ref=" BidMidAsk.model "/>
    <xsd:element name="spreadValue" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="definition" type=" AssetReference " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VolatilityMatrix

[Table of contents]

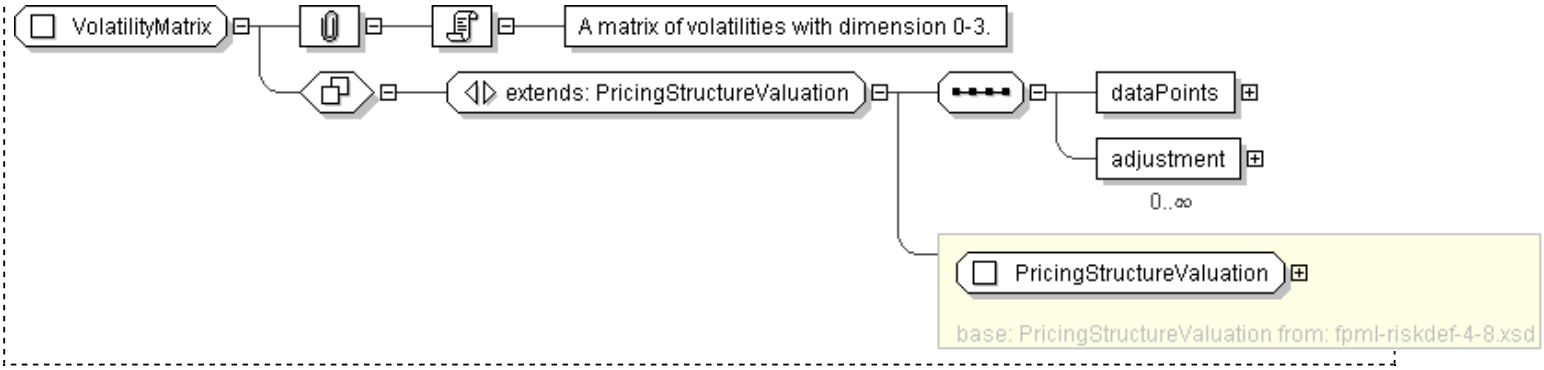
Super-types:	PricingStructureValuation < VolatilityMatrix (by extension)
Sub-types:	None

Name	VolatilityMatrix
Used by (from the same schema document)	Element volatilityMatrixValuation
Abstract	no
Documentation	A matrix of volatilities with dimension 0-3.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
 'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the
   Valuation occurs within a ValuationSet, this value is optional and is defaulted from
   the ValuationSet. If this value occurs in both places, the lower level value (i.e. the
   one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will
   align with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the
   base date. The difference between the baseDate and the spotDate is termed the
   settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained.
   Often the same as the baseDate, but sometimes the pricing input may be \"rolled
   forward\", in which input data from one date is used to generate a curve for a later
   date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <dataPoints> MultiDimensionalPricingData </dataPoints> [1]
   'The raw volatility matrix data, expressed as a multi-dimensional array.'

   <adjustment> ParametricAdjustment </adjustment> [0..*]
   'An adjustment factor, such as for vol smile/skew.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VolatilityMatrix">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructureValuation ">
      <xsd:sequence>
        <xsd:element name="dataPoints" type=" MultiDimensionalPricingData "/>
        <xsd:element name="adjustment" type=" ParametricAdjustment " minOccurs="0"
        maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VolatilityRepresentation

[Table of contents]

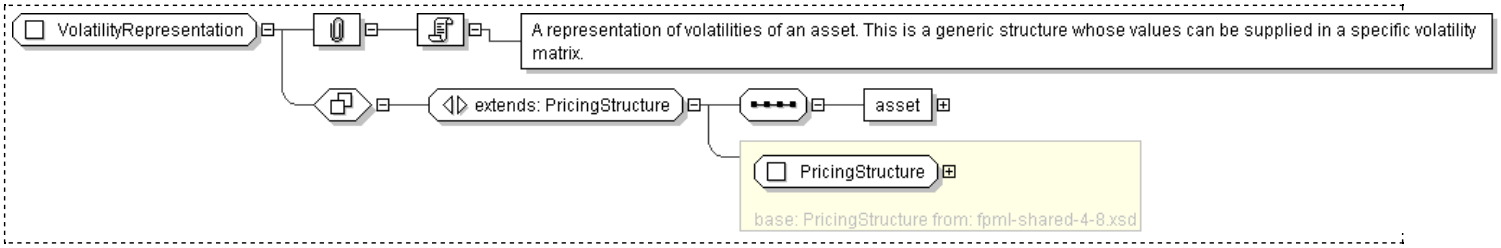
Super-types:	PricingStructure < VolatilityRepresentation (by extension)
Sub-types:	None

Name	VolatilityRepresentation
Used by (from the same schema document)	Element volatilityRepresentation
Abstract	no
Documentation	A representation of volatilities of an asset. This is a generic structure whose values can be supplied in a specific volatility matrix.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates asset class).'

   <asset> AnyAssetReference </asset> [1]
   'A reference to the asset whose volatility is modeled.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VolatilityRepresentation">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructure ">
      <xsd:sequence>
        <xsd:element name="asset" type=" AnyAssetReference "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: YieldCurve

[Table of contents]

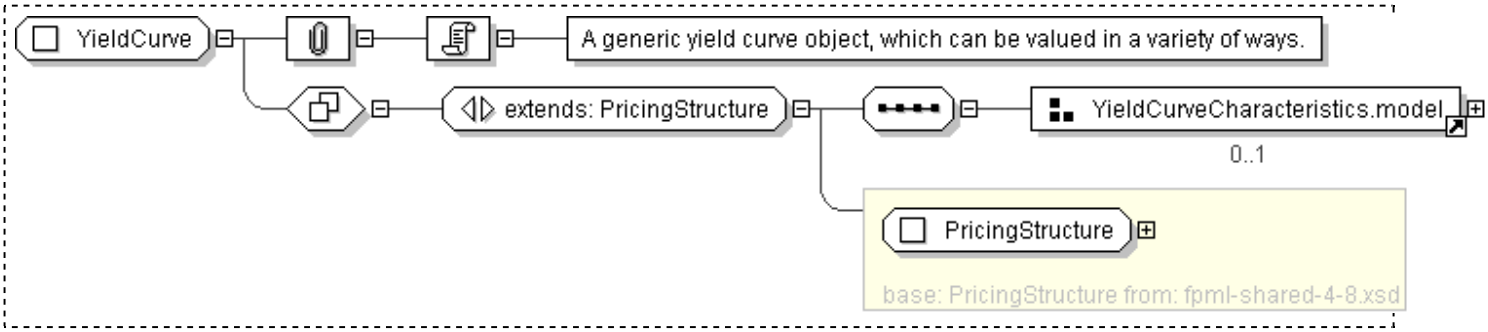
Super-types:	PricingStructure < YieldCurve (by extension)
Sub-types:	None

Name	YieldCurve
Used by (from the same schema document)	Element yieldCurve
Abstract	no
Documentation	A generic yield curve object, which can be valued in a variety of ways.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the
   Interes Rates asset class).'

   Start Group: YieldCurveCharacteristics.model [0..1]
      <algorithm> xsd:string </algorithm> [0..1]
      <forecastRateIndex> ForecastRateIndex </forecastRateIndex> [0..1]
   End Group: YieldCurveCharacteristics.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="YieldCurve">
   <xsd:complexContent>
      <xsd:extension base=" PricingStructure ">
         <xsd:sequence>
            <xsd:group ref=" YieldCurveCharacteristics.model " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: YieldCurveValuation

[Table of contents]

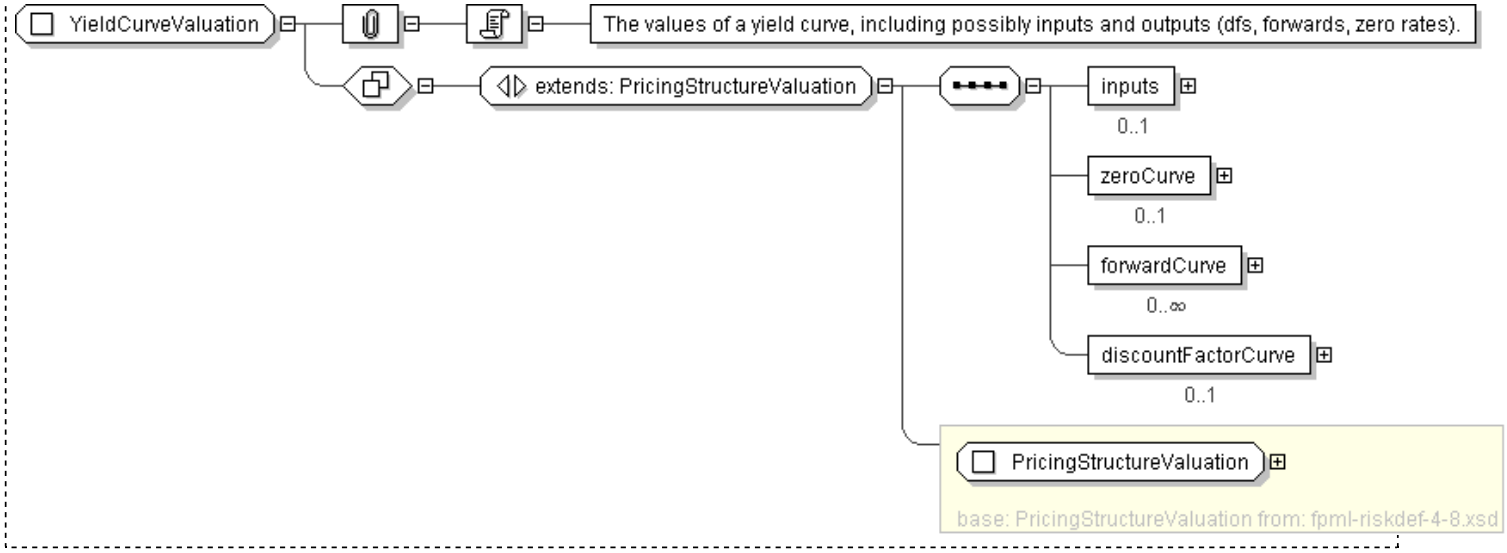
Super-types:	PricingStructureValuation < YieldCurveValuation (by extension)
Sub-types:	None

Name	YieldCurveValuation
Used by (from the same schema document)	Element yieldCurveValuation
Abstract	no
Documentation	The values of a yield curve, including possibly inputs and outputs (dfs, forwards, zero rates).

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
 'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation
   occurs within a ValuationSet, this value is optional and is defaulted from the ValuationSet.
   If this value occurs in both places, the lower level value (i.e. the one here) overrides that
   in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align
   with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base
   date. The difference between the baseDate and the spotDate is termed the settlement lag, and
   is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often
   the same as the baseDate, but sometimes the pricing input may be \"rolled forward\", in which
   input data from one date is used to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

   <inputs> QuotedAssetSet </inputs> [0..1]
   <zeroCurve> ZeroRateCurve </zeroCurve> [0..1]
   'A curve of zero rates.'

   <forwardCurve> ForwardRateCurve </forwardCurve> [0..*]
   'A curve of forward rates.'

   <discountFactorCurve> TermCurve </discountFactorCurve> [0..1]
   'A curve of discount factors.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="YieldCurveValuation">
  <xsd:complexContent>
    <xsd:extension base=" PricingStructureValuation ">
      <xsd:sequence>
        <xsd:element name="inputs" type=" QuotedAssetSet " minOccurs="0"/>
        <xsd:element name="zeroCurve" type=" ZeroRateCurve " minOccurs="0"/>
        <xsd:element name="forwardCurve" type=" ForwardRateCurve " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="discountFactorCurve" type=" TermCurve " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ZeroRateCurve

[Table of contents]

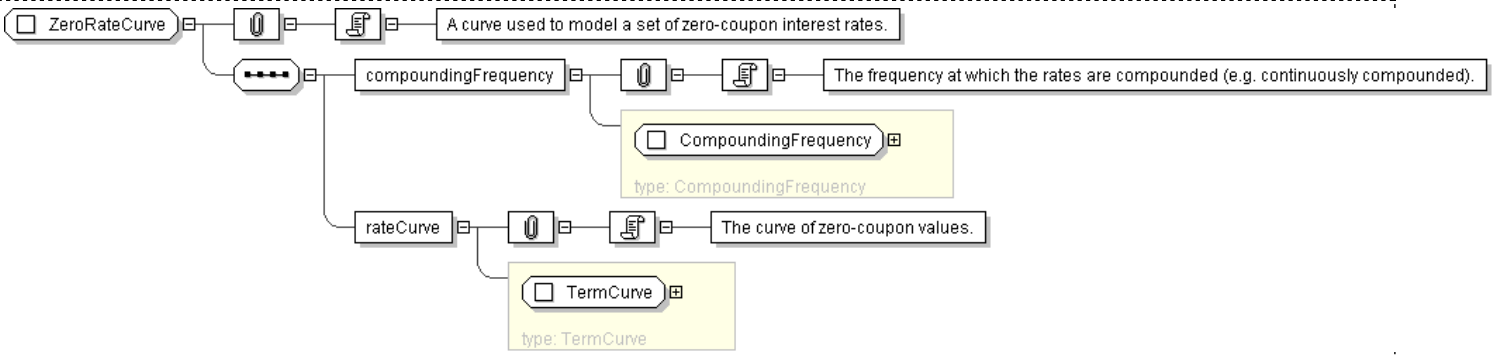
Super-types:	None
Sub-types:	None

Name	ZeroRateCurve
Used by (from the same schema document)	Complex Type YieldCurveValuation
Abstract	no
Documentation	A curve used to model a set of zero-coupon interest rates.

XML Instance Representation

```
<...>
   <compoundingFrequency> CompoundingFrequency </compoundingFrequency> [1]
    'The frequency at which the rates are compounded (e.g. continuously compounded).'

   <rateCurve> TermCurve </rateCurve> [1]
    'The curve of zero-coupon values.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ZeroRateCurve">
   <xsd:sequence>
      <xsd:element name="compoundingFrequency" type=" CompoundingFrequency "/>
      <xsd:element name="rateCurve" type=" TermCurve "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: AdditionalData
  - Complex Type: ConversationId
  - Complex Type: Message
  - Complex Type: MessageAddress
  - Complex Type: MessageHeader
  - Complex Type: MessageId
  - Complex Type: MessageRejected
  - Complex Type: NotificationMessage
  - Complex Type: NotificationMessageHeader
  - Complex Type: PartyMessageInformation
  - Complex Type: ProblemLocation
  - Complex Type: Reason
  - Complex Type: ReasonCode
  - Complex Type: RequestMessage
  - Complex Type: RequestMessageHeader
  - Complex Type: RequestTradeStatus
  - Complex Type: ResponseMessage
  - Complex Type: ResponseMessageHeader
  - Complex Type: TradeAlreadyCancelled
  - Complex Type: TradeAlreadySubmitted
  - Complex Type: TradeAlreadyTerminated
  - Complex Type: TradeErrorResponse
  - Complex Type: TradeNotFound
  - Complex Type: TradeStatus
  - Complex Type: TradeStatusItem
  - Complex Type: TradeStatusValue
  - Model Group: Exception.model
  - Model Group: MessageHeader.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema imports schema(s) from the following namespace(s):
	  ◦ *http://www.w3.org/2000/09/xmldsig#* (at xmldsig-core-schema.xsd)
	• This schema includes components from the following schema document(s):
	  ◦ fpml-doc-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml-annotation	http://www.fpml.org/annotation
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="xmldsig-
   core-schema.xsd"/>
   <xsd:include schemaLocation="fpml-doc-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress
Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Exception.model

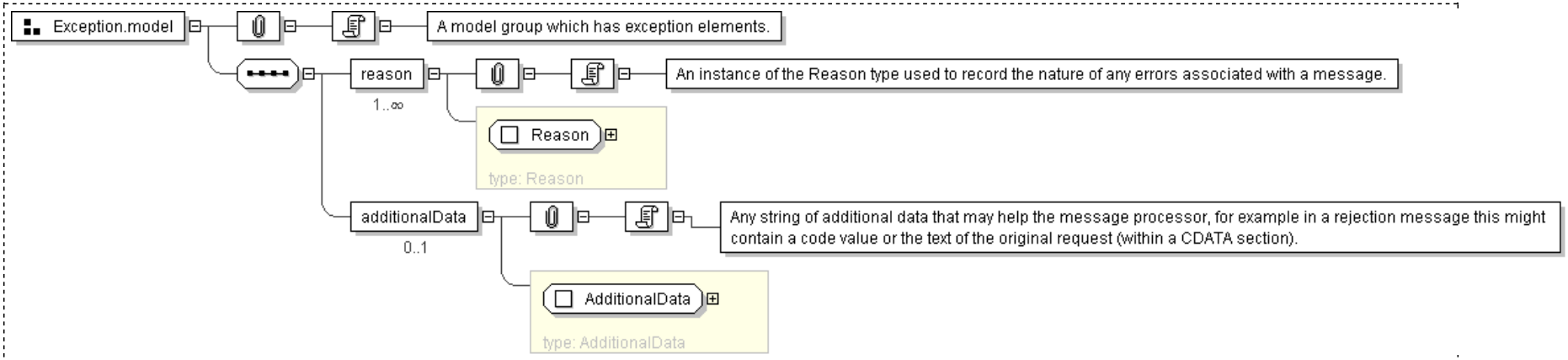
[Table of contents]

Name	Exception.model
Used by (from the same schema document)	Complex Type MessageRejected
Documentation	A model group which has exception elements.

XML Instance Representation

```
<reason> Reason </reason> [1..*]
'An instance of the Reason type used to record the nature of any errors associated with a message.'

<additionalData> AdditionalData </additionalData> [0..1]
'Any string of additional data that may help the message processor, for example in a rejection message this might contain a code value or the text of the original request (within a CDATA section).'
```

Diagram



Schema Component Representation

```
<xsd:group name="Exception.model">
  <xsd:sequence>
    <xsd:element name="reason" type=" Reason " maxOccurs="unbounded"/>
    <xsd:element name="additionalData" type=" AdditionalData " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: MessageHeader.model

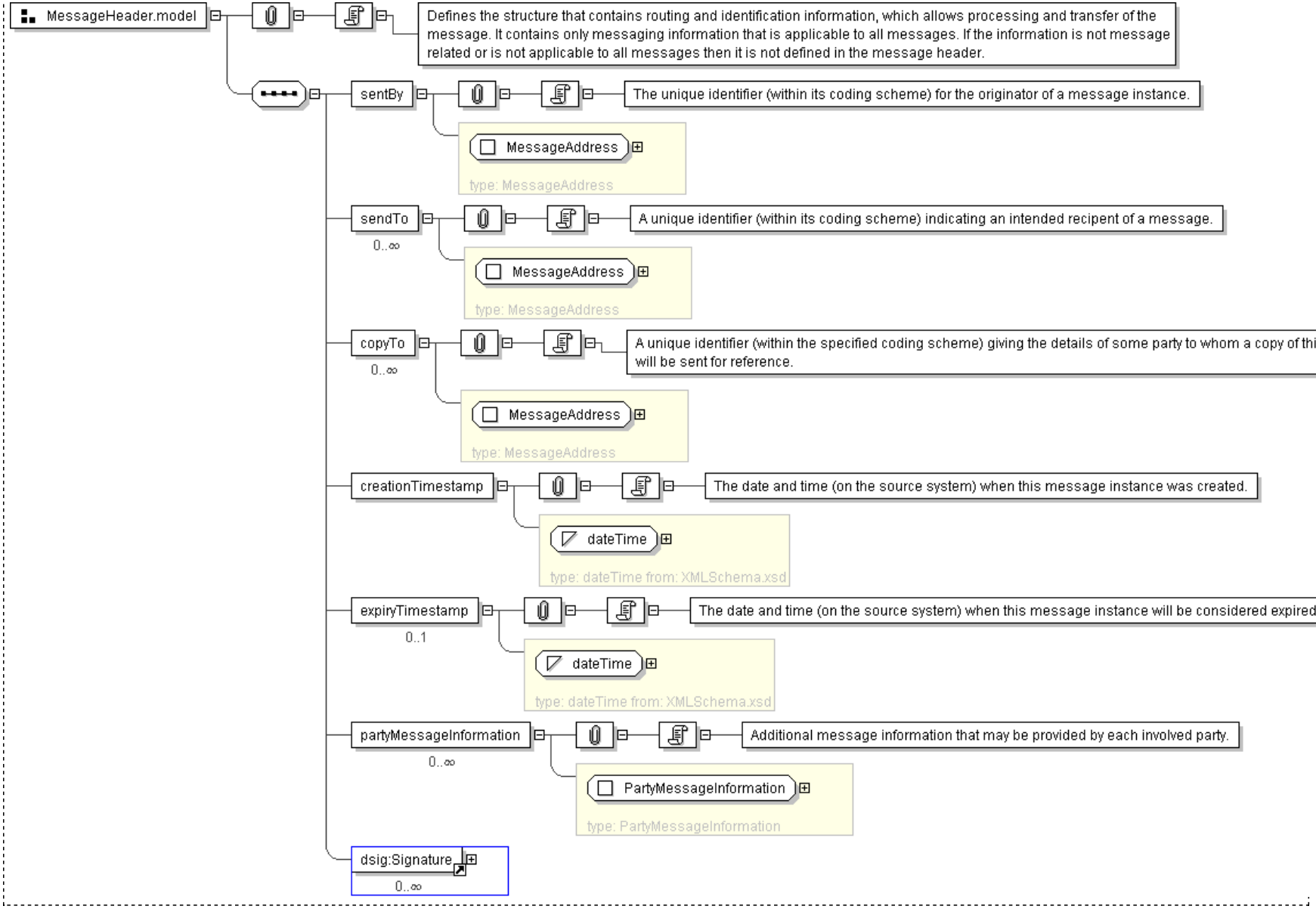
[Table of contents]

Name	MessageHeader.model
Used by (from the same schema document)	Complex Type NotificationMessageHeader , Complex Type RequestMessageHeader , Complex Type ResponseMessageHeader
Documentation	Defines the structure that contains routing and identification information, which allows processing and transfer of the message. It contains only messaging information that is applicable to all messages. If the information is not message related or is not applicable to all messages then it is not defined in the message header.

XML Instance Representation

```
<sentBy> MessageAddress </sentBy> [1]
'The unique identifier (within its coding scheme) for the originator of a message instance.'

<sendTo> MessageAddress </sendTo> [0..*]
'A unique identifier (within its coding scheme) indicating an intended recipent of a message.'

<copyTo> MessageAddress </copyTo> [0..*]
'A unique identifier (within the specified coding scheme) giving the details of some party to whom a copy of this message will be sent
for reference.'

<creationTimestamp> xsd:dateTime </creationTimestamp> [1]
'The date and time (on the source system) when this message instance was created.'

<expiryTimestamp> xsd:dateTime </expiryTimestamp> [0..1]
'The date and time (on the source system) when this message instance will be considered expired.'

<partyMessageInformation> PartyMessageInformation </partyMessageInformation> [0..*]
'Additional message information that may be provided by each involved party.'

<dsig:Signature> ... </dsig:Signature> [0..*]
```

Diagram



Schema Component Representation

```
<xsd:group name="MessageHeader.model">
  <xsd:sequence>
    <xsd:element name="sentBy" type=" MessageAddress "/>
    <xsd:element name="sendTo" type=" MessageAddress " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="copyTo" type=" MessageAddress " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="creationTimestamp" type=" xsd:dateTime "/>
    <xsd:element name="expiryTimestamp" type=" xsd:dateTime " minOccurs="0"/>
    <xsd:element name="partyMessageInformation" type=" PartyMessageInformation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref=" dsig:Signature " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdditionalData

[Table of contents]

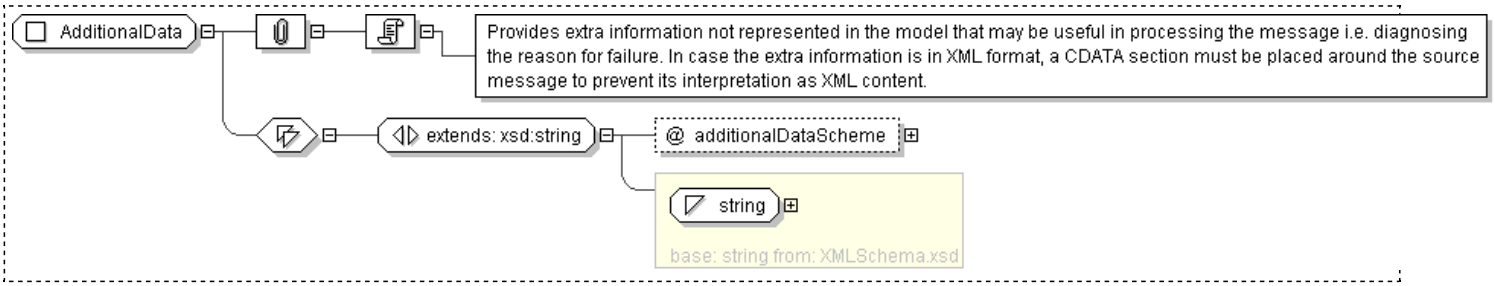
Super-types:	xsd:string < AdditionalData (by extension)
Sub-types:	None

Name	AdditionalData
Used by (from the same schema document)	Complex Type Reason , Model Group Exception.model
Abstract	no
Documentation	Provides extra information not represented in the model that may be useful in processing the message i.e. diagnosing the reason for failure. In case the extra information is in XML format, a CDATA section must be placed around the source message to prevent its interpretation as XML content.

XML Instance Representation

```
<...
 additionalDataScheme=" xsd:anyURI [0..1]">
   xsd:string
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdditionalData">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:string ">
      <xsd:attribute name="additionalDataScheme" type=" xsd:anyURI " use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ConversationId

[Table of contents]

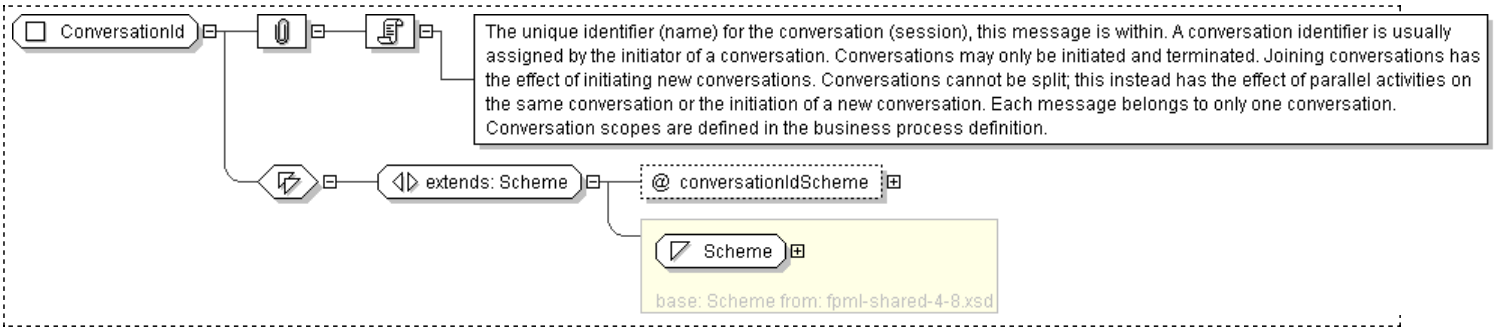
Super-types:	Scheme < ConversationId (by extension)
Sub-types:	None

Name	ConversationId
Used by (from the same schema document)	Complex Type MessageHeader
Abstract	no
Documentation	The unique identifier (name) for the conversation (session), this message is within. A conversation identifier is usually assigned by the initiator of a conversation. Conversations may only be initiated and terminated. Joining conversations has the effect of initiating new conversations. Conversations cannot be split; this instead has the effect of parallel activities on the same conversation or the initiation of a new conversation. Each message belongs to only one conversation. Conversation scopes are defined in the business process definition.

XML Instance Representation

```
<...
 conversationIdScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ConversationId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="conversationIdScheme" type=" xsd:anyURI " use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Message

[Table of contents]

Super-types: Document < Message (by extension)

Sub-types:

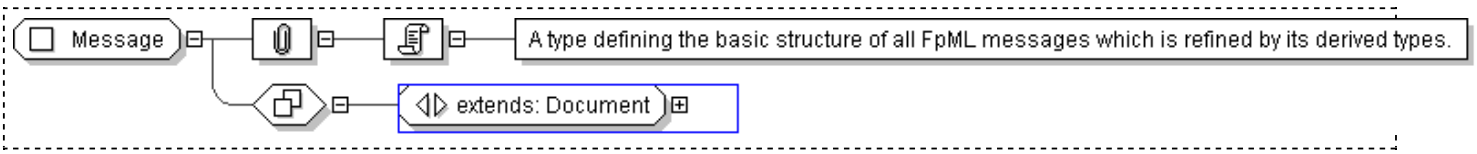
- NotificationMessage (by extension)
  - MessageRejected (by extension)
- RequestMessage (by extension)
  - RequestTradeStatus (by extension)
- ResponseMessage (by extension)
  - TradeNotFound (by extension)
  - TradeStatus (by extension)
  - TradeErrorResponse (by extension)
    - TradeAlreadyCancelled (by extension)
    - TradeAlreadyTerminated (by extension)
  - TradeAlreadySubmitted (by extension)

Name	Message
Abstract	yes
Documentation	A type defining the basic structure of all FpML messages which is refined by its derived types.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 "/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Message" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Document "/>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MessageAddress

[Table of contents]

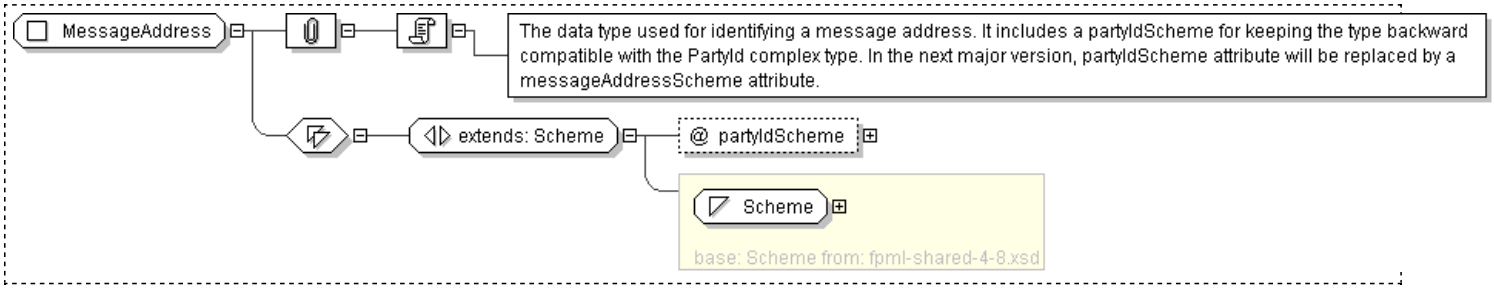
Super-types:	Scheme < MessageAddress (by extension)
Sub-types:	None

Name	MessageAddress
Used by (from the same schema document)	Model Group MessageHeader.model , Model Group MessageHeader.model , Model Group MessageHeader.model
Abstract	no
Documentation	The data type used for identifying a message address. It includes a partyIdScheme for keeping the type backward compatible with the PartyId complex type. In the next major version, partyIdScheme attribute will be replaced by a messageAddressScheme attribute.

XML Instance Representation

```
<...
 partyIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MessageAddress">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="partyIdScheme" type=" xsd:anyURI " default="http://www.fpml.org/ext/iso9362"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MessageHeader

[Table of contents]

Super-types: None

Sub-types:

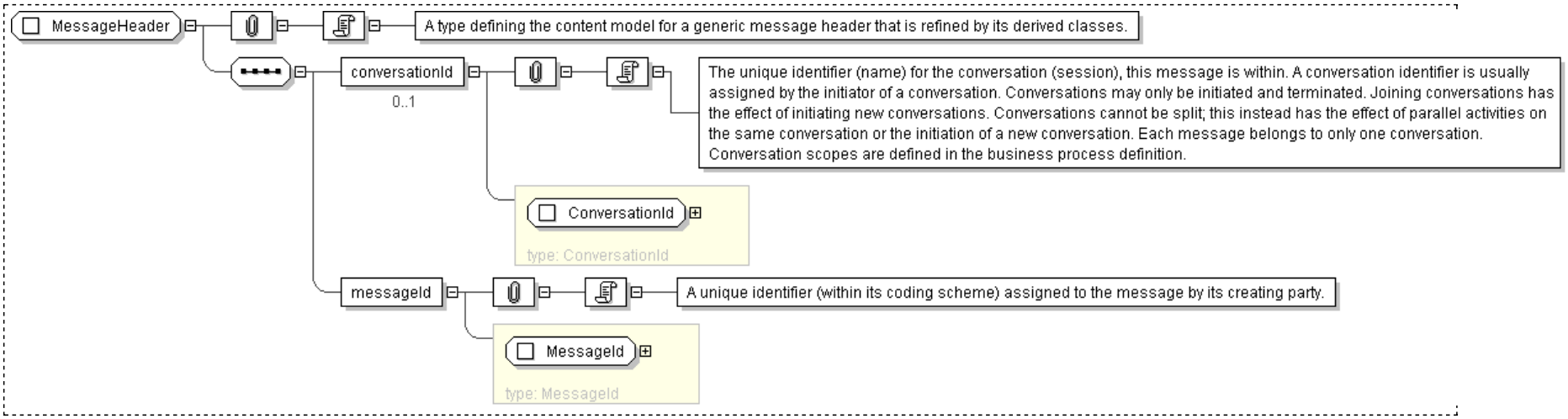
- NotificationMessageHeader (by extension)
- RequestMessageHeader (by extension)
- ResponseMessageHeader (by extension)

Name	MessageHeader
Abstract	yes
Documentation	A type defining the content model for a generic message header that is refined by its derived classes.

XML Instance Representation

```
<...>
   <conversationId> ConversationId </conversationId> [0..1]
   'The unique identifier (name) for the conversation (session), this message is within. A conversation identifier is usually assigned by
   the initiator of a conversation. Conversations may only be initiated and terminated. Joining conversations has the effect of initiating
   new conversations. Conversations cannot be split; this instead has the effect of parallel activities on the same conversation or the
   initiation of a new conversation. Each message belongs to only one conversation. Conversation scopes are defined in the business
   process definition.'

   <messageId> MessageId </messageId> [1]
   'A unique identifier (within its coding scheme) assigned to the message by its creating party.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MessageHeader" abstract="true">
   <xsd:sequence>
      <xsd:element name="conversationId" type=" ConversationId " minOccurs="0"/>
      <xsd:element name="messageId" type=" MessageId "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MessageId

[Table of contents]

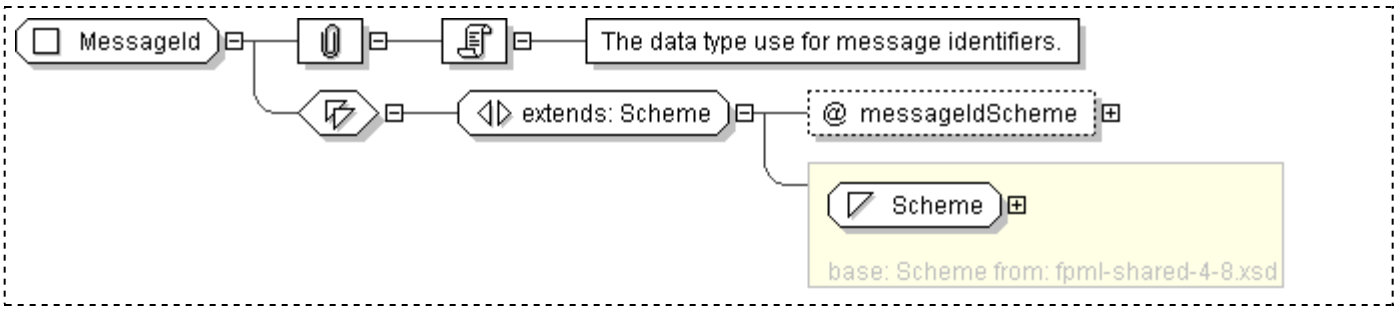
Super-types:	Scheme < MessageId (by extension)
Sub-types:	None

Name	MessageId
Used by (from the same schema document)	Complex Type MessageHeader , Complex Type NotificationMessageHeader , Complex Type ResponseMessageHeader
Abstract	no
Documentation	The data type use for message identifiers.

XML Instance Representation

```
<...
 messageIdScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MessageId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="messageIdScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MessageRejected

[Table of contents]

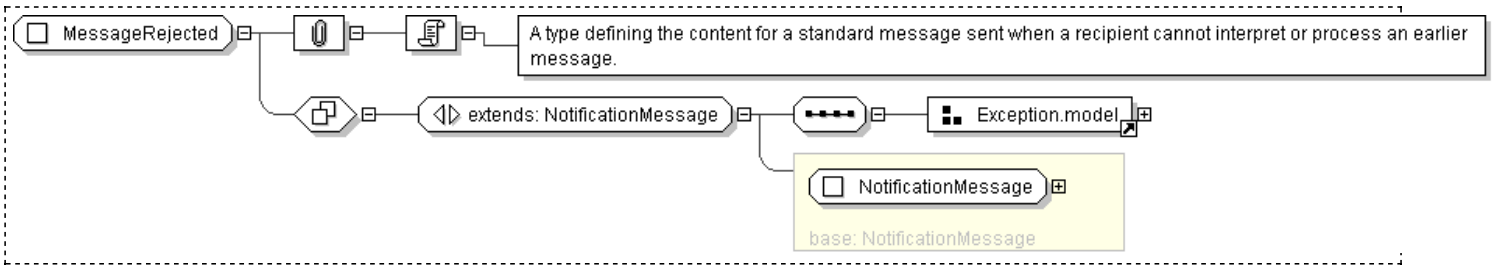
Super-types:	Document < Message (by extension) < NotificationMessage (by extension) < MessageRejected (by extension)
Sub-types:	None

Name	MessageRejected
Abstract	no
Documentation	A type defining the content for a standard message sent when a recipient cannot interpret or process an earlier message.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
 changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
 the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <reason> Reason </reason> [1..*]
   'An instance of the Reason type used to record the nature of any errors associated with a message.'

   <additionalData> AdditionalData </additionalData> [0..1]
   'Any string of additional data that may help the message processor, for example in a rejection message
   this might contain a code value or the text of the original request (within a CDATA section).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MessageRejected">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:group ref=" Exception.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotificationMessage

[Table of contents]

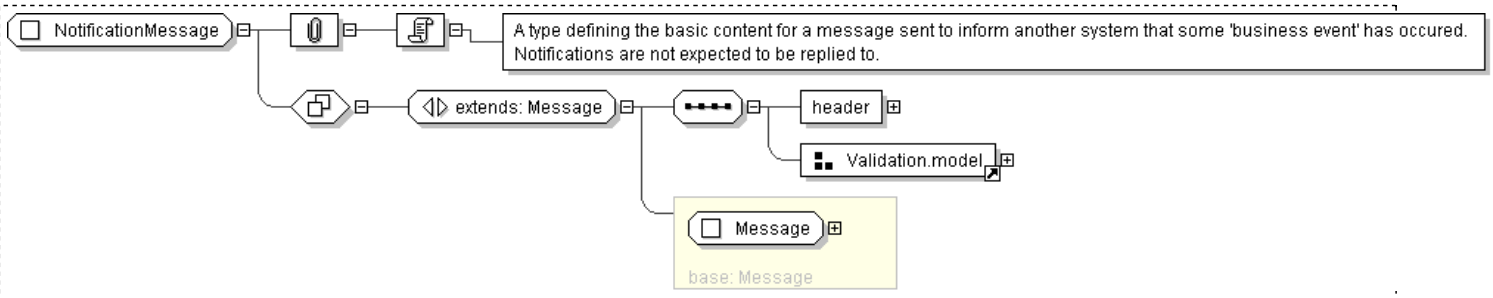
Super-types:	Document < Message (by extension) < NotificationMessage (by extension)
Sub-types:	• MessageRejected (by extension)

Name	NotificationMessage
Abstract	yes
Documentation	A type defining the basic content for a message sent to inform another system that some 'business event' has occured. Notifications are not expected to be replied to.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the
build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples
within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between
releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotificationMessage" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Message ">
      <xsd:sequence>
        <xsd:element name="header" type=" NotificationMessageHeader "/>
        <xsd:group ref=" Validation.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotificationMessageHeader

[Table of contents]

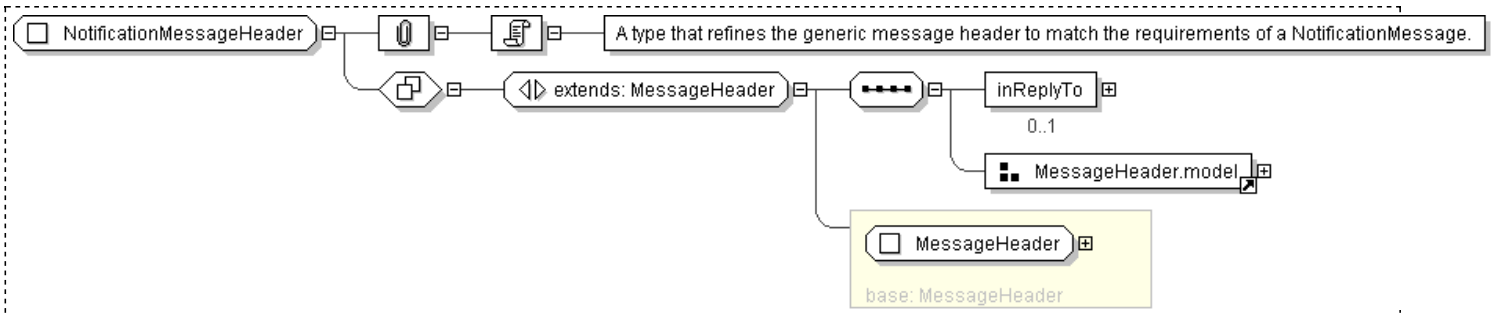
Super-types:	MessageHeader < NotificationMessageHeader (by extension)
Sub-types:	None

Name	NotificationMessageHeader
Used by (from the same schema document)	Complex Type NotificationMessage
Abstract	no
Documentation	A type that refines the generic message header to match the requirements of a NotificationMessage.

XML Instance Representation

```
<...>
  <conversationId> ConversationId </conversationId> [0..1]
  'The unique identifier (name) for the conversation (session), this message is within. A conversation
  identifier is usually assigned by the initiator of a conversation. Conversations may only be initiated
  and terminated. Joining conversations has the effect of initiating new conversations. Conversations
  cannot be split; this instead has the effect of parallel activities on the same conversation or the
  initiation of a new conversation. Each message belongs to only one conversation. Conversation scopes are
  defined in the business process definition.'

  <messageId> MessageId </messageId> [1]
  'A unique identifier (within its coding scheme) assigned to the message by its creating party.'

  <inReplyTo> MessageId </inReplyTo> [0..1]
  'A copy of the unique message identifier (within it own coding scheme) to which this message is
  responding.'

  <sentBy> MessageAddress </sentBy> [1]
  'The unique identifier (within its coding scheme) for the originator of a message instance.'

  <sendTo> MessageAddress </sendTo> [0..*]
  'A unique identifier (within its coding scheme) indicating an intended recipent of a message.'

  <copyTo> MessageAddress </copyTo> [0..*]
  'A unique identifier (within the specified coding scheme) giving the details of some party to whom a
  copy of this message will be sent for reference.'

  <creationTimestamp> xsd:dateTime </creationTimestamp> [1]
  'The date and time (on the source system) when this message instance was created.'

  <expiryTimestamp> xsd:dateTime </expiryTimestamp> [0..1]
  'The date and time (on the source system) when this message instance will be considered expired.'

  <partyMessageInformation> PartyMessageInformation </partyMessageInformation> [0..*]
  'Additional message information that may be provided by each involved party.'

  <dsig:Signature> ... </dsig:Signature> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotificationMessageHeader">
  <xsd:complexContent>
    <xsd:extension base=" MessageHeader ">
      <xsd:sequence>
        <xsd:element name="inReplyTo" type=" MessageId " minOccurs="0"/>
        <xsd:group ref=" MessageHeader.model "/>
      </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyMessageInformation

[Table of contents]

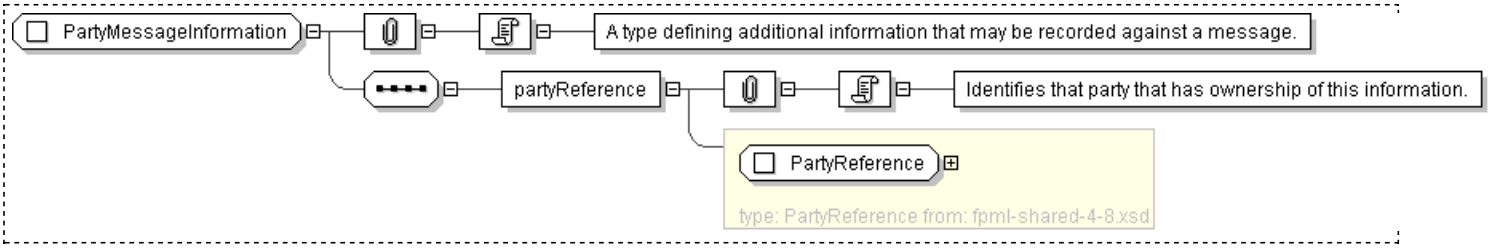
Super-types:	None
Sub-types:	None

Name	PartyMessageInformation
Used by (from the same schema document)	Model Group MessageHeader.model
Abstract	no
Documentation	A type defining additional information that may be recorded against a message.

XML Instance Representation

```
<...>
  <partyReference> PartyReference </partyReference> [1]
   'Identifies that party that has ownership of this information.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyMessageInformation">
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProblemLocation

[Table of contents]

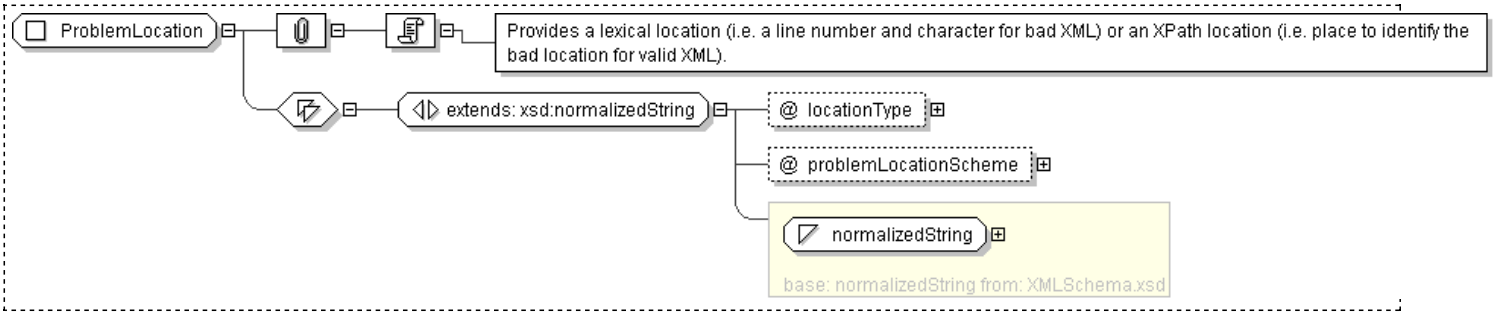
Super-types:	xsd:normalizedString < ProblemLocation (by extension)
Sub-types:	None

Name	ProblemLocation
Used by (from the same schema document)	Complex Type Reason
Abstract	no
Documentation	Provides a lexical location (i.e. a line number and character for bad XML) or an XPath location (i.e. place to identify the bad location for valid XML).

XML Instance Representation

```
<...
 locationType=" xsd:token [0..1]
 'The value of the locationType attribute defines which type of location has been given. It may take the values \'lexical\' or \'xpath\'.'

 "
 problemLocationScheme=" xsd:anyURI [0..1]
 'DEPRECATED. It will be removed in FpML 5.0. New implementations are encouraged to use the locationType attribute.'

 ">
   xsd:normalizedString
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProblemLocation">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:normalizedString ">
      <xsd:attribute name="locationType" type=" xsd:token "/>
      <xsd:attribute name="problemLocationScheme" type=" xsd:anyURI " deprecated="true" deprecatedReason="This
      attribute was introduced by mistake in FpML 4.2. instead of keeping the locationType attribute."/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Reason

[Table of contents]

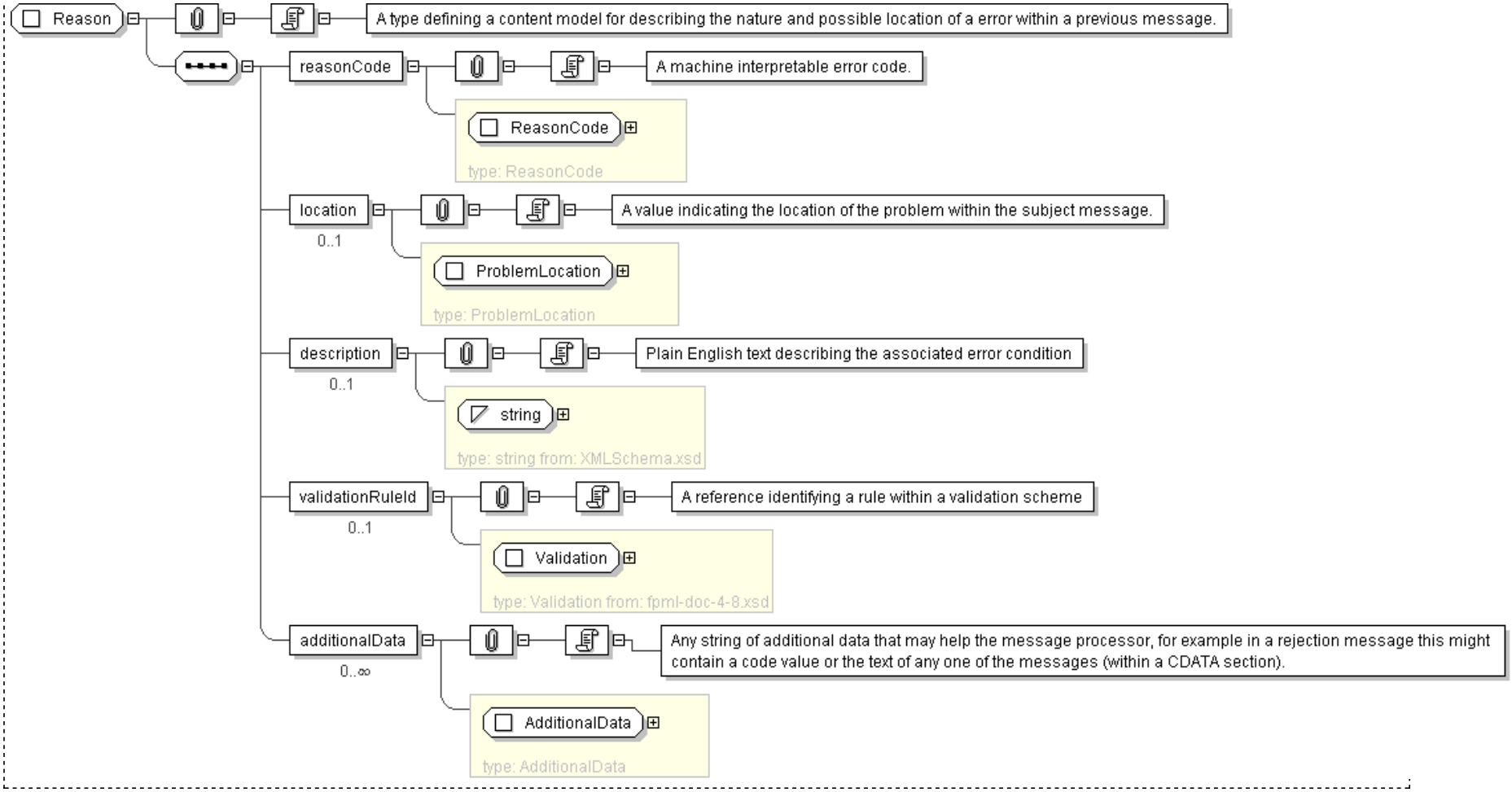
Super-types:	None
Sub-types:	None

Name	Reason
Used by (from the same schema document)	Model Group Exception.model
Abstract	no
Documentation	A type defining a content model for describing the nature and possible location of a error within a previous message.

XML Instance Representation

```
<...>
   <reasonCode> ReasonCode </reasonCode> [1]
   'A machine interpretable error code.'

   <location> ProblemLocation </location> [0..1]
   'A value indicating the location of the problem within the subject message.'

   <description> xsd:string </description> [0..1]
   'Plain English text describing the associated error condition'

   <validationRuleId> Validation </validationRuleId> [0..1]
   'A reference identifying a rule within a validation scheme'

   <additionalData> AdditionalData </additionalData> [0..*]
   'Any string of additional data that may help the message processor, for example in a rejection message this might contain a
   code value or the text of any one of the messages (within a CDATA section).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Reason">
  <xsd:sequence>
    <xsd:element name="reasonCode" type=" ReasonCode "/>
    <xsd:element name="location" type=" ProblemLocation " minOccurs="0"/>
    <xsd:element name="description" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="validationRuleId" type=" Validation " minOccurs="0"/>
    <xsd:element name="additionalData" type=" AdditionalData " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReasonCode

[Table of contents]

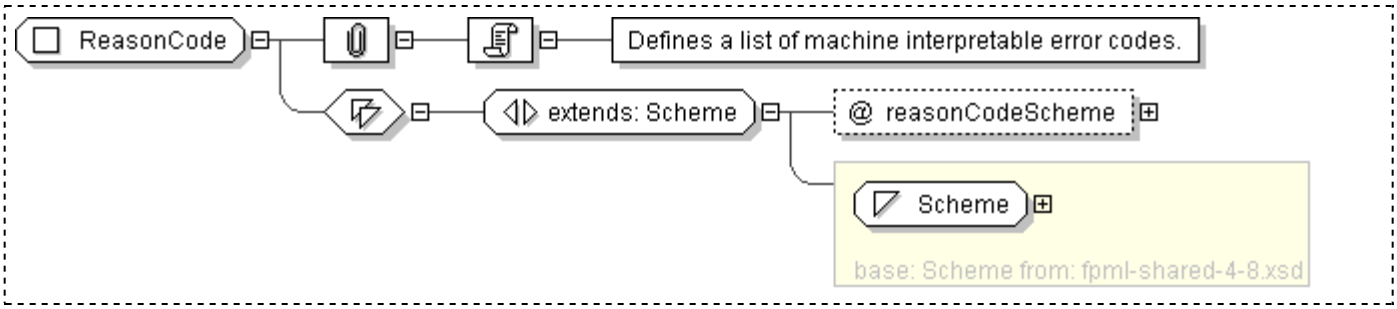
Super-types:	Scheme < ReasonCode (by extension)
Sub-types:	None

Name	ReasonCode
Used by (from the same schema document)	Complex Type Reason
Abstract	no
Documentation	Defines a list of machine interpretable error codes.

XML Instance Representation

```
<...
 reasonCodeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReasonCode">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="reasonCodeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/reason-code"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestMessage

[Table of contents]

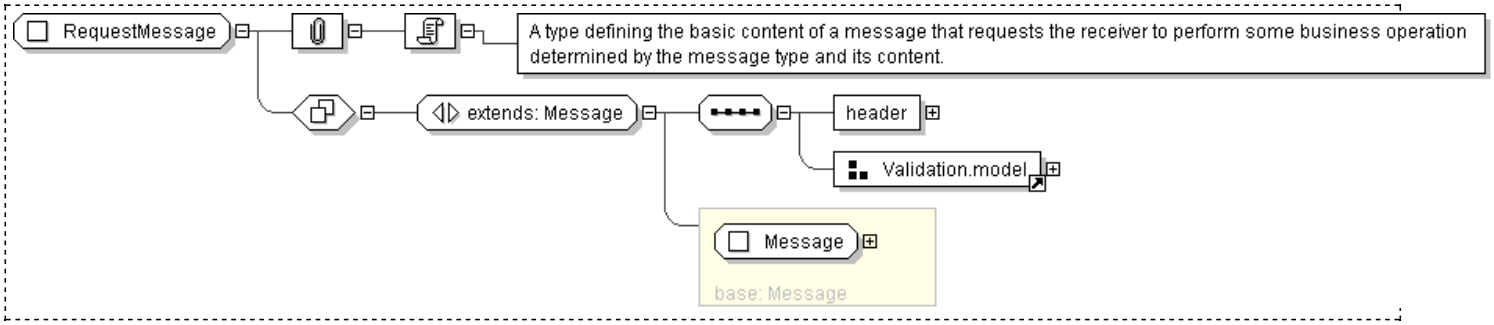
Super-types:	Document < Message (by extension) < RequestMessage (by extension)
Sub-types:	• RequestTradeStatus (by extension)

Name	RequestMessage
Abstract	yes
Documentation	A type defining the basic content of a message that requests the receiver to perform some business operation determined by the message type and its content.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestMessage" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Message ">
      <xsd:sequence>
        <xsd:element name="header" type=" RequestMessageHeader "/>
        <xsd:group ref=" Validation.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestMessageHeader

[Table of contents]

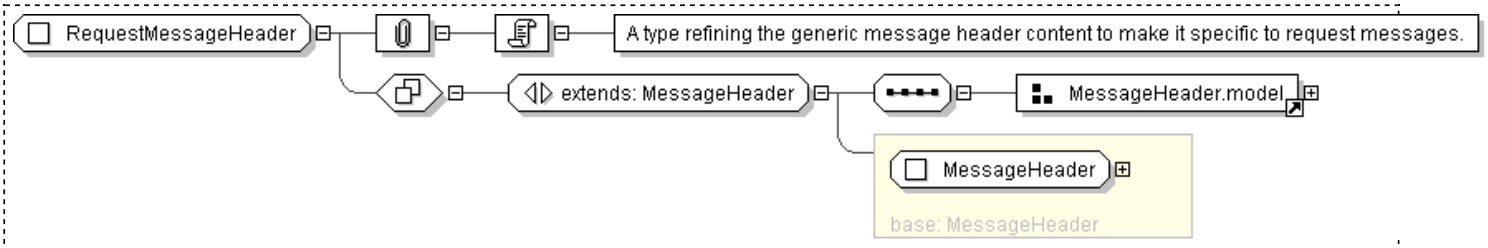
Super-types:	MessageHeader < RequestMessageHeader (by extension)
Sub-types:	None

Name	RequestMessageHeader
Used by (from the same schema document)	Complex Type RequestMessage
Abstract	no
Documentation	A type refining the generic message header content to make it specific to request messages.

XML Instance Representation

```
<...>
   <conversationId> ConversationId </conversationId> [0..1]
   'The unique identifier (name) for the conversation (session), this message is within. A
   conversation identifier is usually assigned by the initiator of a conversation. Conversations may
   only be initiated and terminated. Joining conversations has the effect of initiating new
   conversations. Conversations cannot be split; this instead has the effect of parallel activities
   on the same conversation or the initiation of a new conversation. Each message belongs to only one
   conversation. Conversation scopes are defined in the business process definition.'

   <messageId> MessageId </messageId> [1]
   'A unique identifier (within its coding scheme) assigned to the message by its creating party.'

   <sentBy> MessageAddress </sentBy> [1]
   'The unique identifier (within its coding scheme) for the originator of a message instance.'

   <sendTo> MessageAddress </sendTo> [0..*]
   'A unique identifier (within its coding scheme) indicating an intended recipent of a message.'

   <copyTo> MessageAddress </copyTo> [0..*]
   'A unique identifier (within the specified coding scheme) giving the details of some party to whom
   a copy of this message will be sent for reference.'

   <creationTimestamp> xsd:dateTime </creationTimestamp> [1]
   'The date and time (on the source system) when this message instance was created.'

   <expiryTimestamp> xsd:dateTime </expiryTimestamp> [0..1]
   'The date and time (on the source system) when this message instance will be considered expired.'

   <partyMessageInformation> PartyMessageInformation </partyMessageInformation> [0..*]
   'Additional message information that may be provided by each involved party.'

   <dsig:Signature> ... </dsig:Signature> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestMessageHeader">
   <xsd:complexContent>
      <xsd:extension base=" MessageHeader ">
         <xsd:sequence>
            <xsd:group ref=" MessageHeader.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestTradeStatus

[Table of contents]

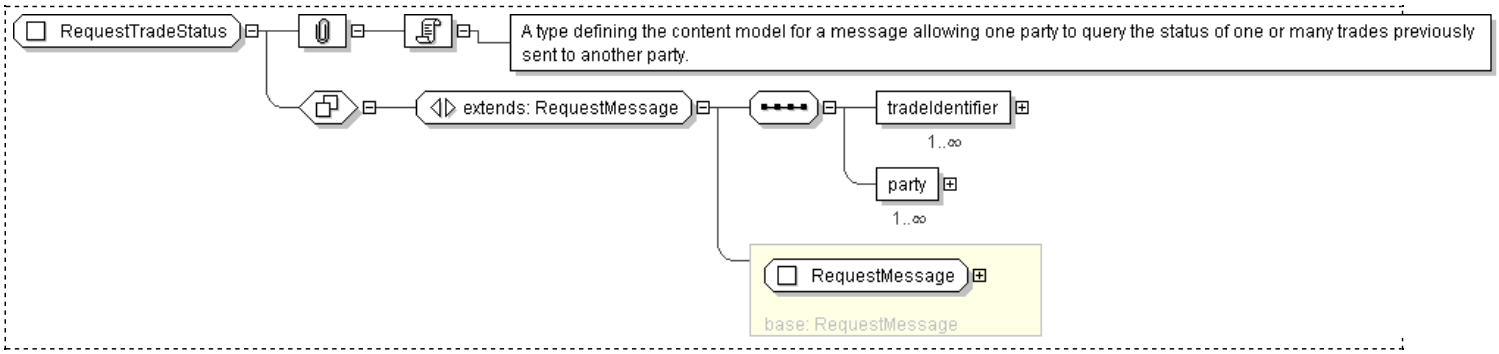
Super-types:	Document < Message (by extension) < RequestMessage (by extension) < RequestTradeStatus (by extension)
Sub-types:	None

Name	RequestTradeStatus
Abstract	no
Documentation	A type defining the content model for a message allowing one party to query the status of one or many trades previously sent to another party.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the
build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples
within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between
releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1..*]
   'An instance of a unique trade identifier.'

   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle.
   For example, the principal parties obligated to make payments from time to time during the term of the trade,
   but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of
   novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places
   within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestTradeStatus">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResponseMessage

[Table of contents]

Super-types: Document < Message (by extension) < ResponseMessage (by extension)

Sub-types:

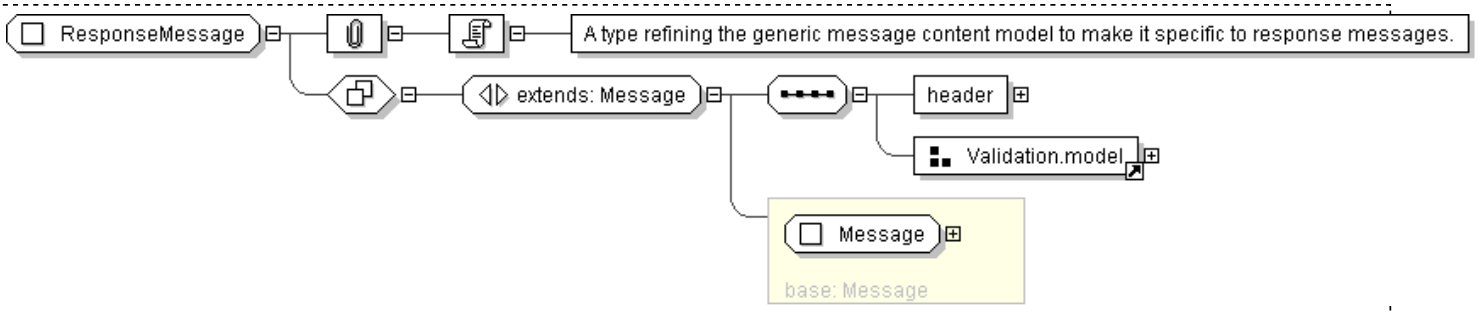
- TradeNotFound (by extension)
- TradeStatus (by extension)
- TradeErrorResponse (by extension)
  - TradeAlreadyCancelled (by extension)
  - TradeAlreadyTerminated (by extension)
- TradeAlreadySubmitted (by extension)

Name	ResponseMessage
Abstract	yes
Documentation	A type refining the generic message content model to make it specific to response messages.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'})
[1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the
 FpML schema and indicates the build number of the schema file. Every time FpML publishes a change
 to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build
 number is incremented. If no changes have been made between releases within a version (i.e. from
 Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResponseMessage" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Message ">
      <xsd:sequence>
        <xsd:element name="header" type=" ResponseMessageHeader "/>
        <xsd:group ref=" Validation.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResponseMessageHeader

[Table of contents]

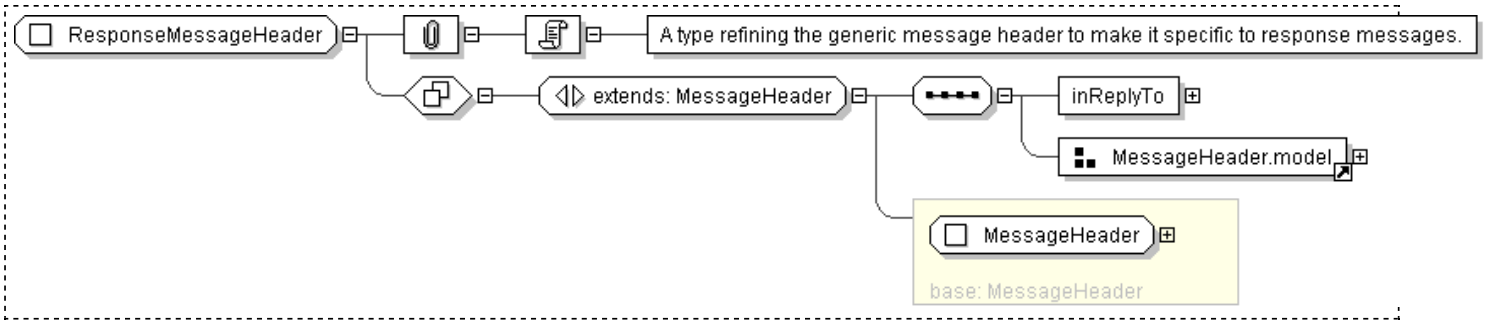
Super-types:	MessageHeader < ResponseMessageHeader (by extension)
Sub-types:	None

Name	ResponseMessageHeader
Used by (from the same schema document)	Complex Type ResponseMessage
Abstract	no
Documentation	A type refining the generic message header to make it specific to response messages.

XML Instance Representation

```
<...>
  <conversationId> ConversationId </conversationId> [0..1]
  'The unique identifier (name) for the conversation (session), this message is within. A
  conversation identifier is usually assigned by the initiator of a conversation. Conversations
  may only be initiated and terminated. Joining conversations has the effect of initiating new
  conversations. Conversations cannot be split; this instead has the effect of parallel activities
  on the same conversation or the initiation of a new conversation. Each message belongs to only
  one conversation. Conversation scopes are defined in the business process definition.'

  <messageId> MessageId </messageId> [1]
  'A unique identifier (within its coding scheme) assigned to the message by its creating party.'

  <inReplyTo> MessageId </inReplyTo> [1]
  'A copy of the unique message identifier (within it own coding scheme) to which this message is
  responding.'

  <sentBy> MessageAddress </sentBy> [1]
  'The unique identifier (within its coding scheme) for the originator of a message instance.'

  <sendTo> MessageAddress </sendTo> [0..*]
  'A unique identifier (within its coding scheme) indicating an intended recipent of a message.'

  <copyTo> MessageAddress </copyTo> [0..*]
  'A unique identifier (within the specified coding scheme) giving the details of some party to
  whom a copy of this message will be sent for reference.'

  <creationTimestamp> xsd:dateTime </creationTimestamp> [1]
  'The date and time (on the source system) when this message instance was created.'

  <expiryTimestamp> xsd:dateTime </expiryTimestamp> [0..1]
  'The date and time (on the source system) when this message instance will be considered
  expired.'

  <partyMessageInformation> PartyMessageInformation </partyMessageInformation> [0..*]
  'Additional message information that may be provided by each involved party.'

  <dsig:Signature> ... </dsig:Signature> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResponseMessageHeader">
  <xsd:complexContent>
    <xsd:extension base=" MessageHeader ">
      <xsd:sequence>
        <xsd:element name="inReplyTo" type=" MessageId "/>
        <xsd:group ref=" MessageHeader.model "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadyCancelled

[Table of contents]

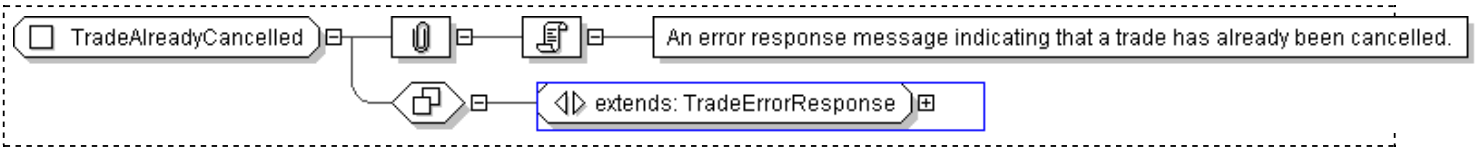
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeErrorResponse (by extension) < TradeAlreadyCancelled (by extension)
Sub-types:	None

Name	TradeAlreadyCancelled
Abstract	no
Documentation	An error response message indicating that a trade has already been cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different party.'

   End Choice
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadyCancelled">
  <xsd:complexContent>
```

```
      <xsd:extension base=" TradeErrorResponse "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadySubmitted

[Table of contents]

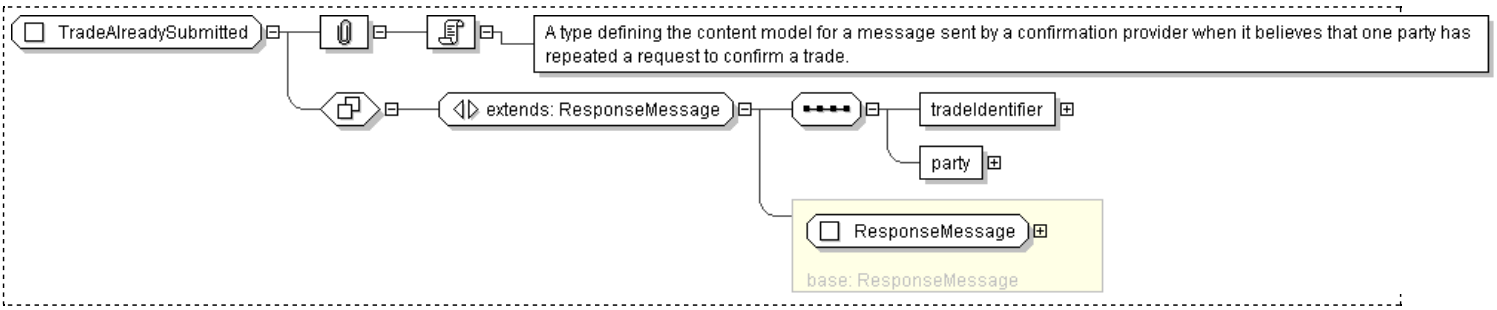
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeAlreadySubmitted (by extension)
Sub-types:	None

Name	TradeAlreadySubmitted
Abstract	no
Documentation	A type defining the content model for a message sent by a confirmation provider when it believes that one party has repeated a request to confirm a trade.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number
stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle.
   For example, the principal parties obligated to make payments from time to time during the term of the trade,
   but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of
   novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places
   within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadySubmitted">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadyTerminated

[Table of contents]

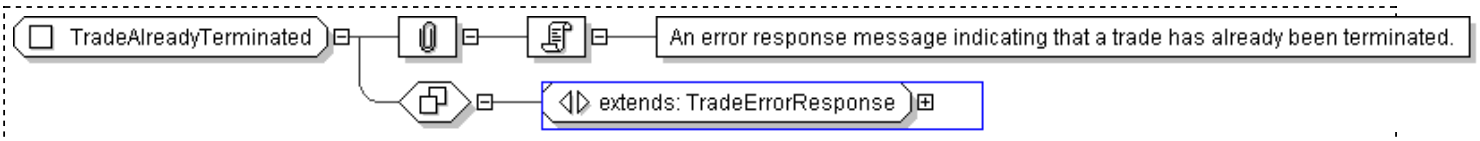
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeErrorResponse (by extension) < TradeAlreadyTerminated (by extension)
Sub-types:	None

Name	TradeAlreadyTerminated
Abstract	no
Documentation	An error response message indicating that a trade has already been terminated.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different party.'

   End Choice
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadyTerminated">
   <xsd:complexContent>
      <xsd:extension base=" TradeErrorResponse "/>
   </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeErrorResponse

[Table of contents]

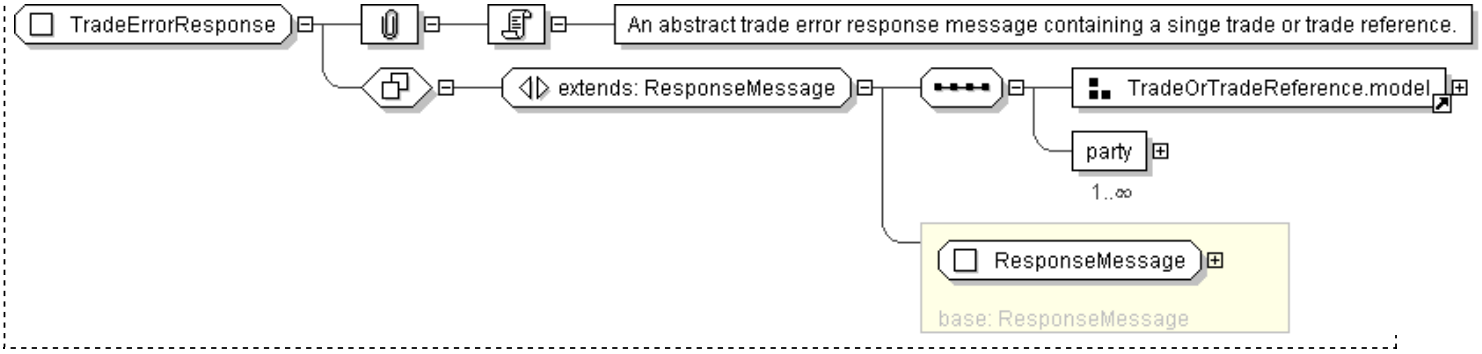
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeErrorResponse (by extension)
Sub-types:	• TradeAlreadyCancelled (by extension) • TradeAlreadyTerminated (by extension)

Name	TradeErrorResponse
Abstract	yes
Documentation	An abstract trade error response message containing a singe trade or trade reference.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more different partyTradeIdentifier elements - each allocated by a different party.'

   End Choice
   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeErrorResponse" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:group ref=" TradeOrTradeReference.model "/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeNotFound

[Table of contents]

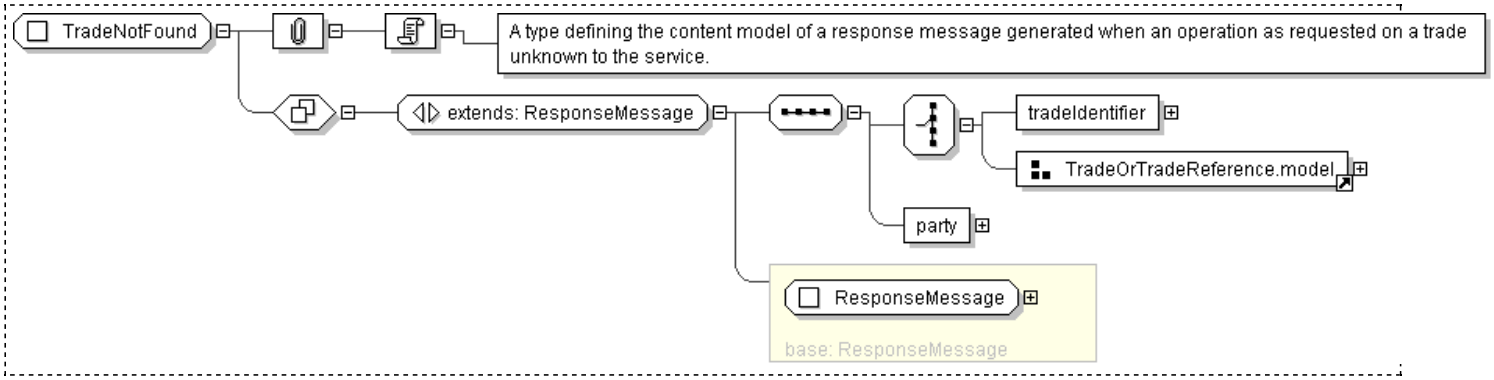
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeNotFound (by extension)
Sub-types:	None

Name	TradeNotFound
Abstract	no
Documentation	A type defining the content model of a response message generated when an operation as requested on a trade unknown to the service.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
 changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
 the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
      'An instance of a unique trade identifier.'

      Start Choice [1]
         <trade> Trade </trade> [1]
         'An element that allows the full details of the trade to be used as a mechanism for identifying the
         trade for which the post-trade event pertains'

         <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
         'A container since an individual trade can be referenced by two or more different
         partyTradeIdentifier elements - each allocated by a different party.'

      End Choice
   End Choice
   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the
   term of the trade, but may include other parties involved in, or incidental to, the trade, such as
   parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML
   roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeNotFound">
```

```
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
          <xsd:group ref=" TradeOrTradeReference.model "/>
        </xsd:choice>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeStatus

[Table of contents]

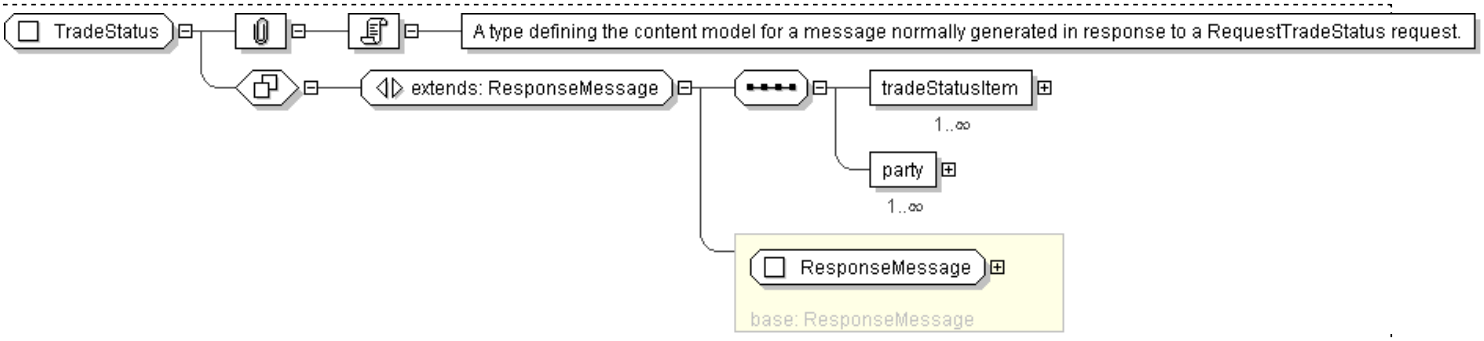
Super-types:	Document < Message (by extension) < ResponseMessage (by extension) < TradeStatus (by extension)
Sub-types:	None

Name	TradeStatus
Abstract	no
Documentation	A type defining the content model for a message normally generated in response to a RequestTradeStatus request.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema,
 validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented.
 If no changes have been made between releases within a version (i.e. from Trial Recommendation to
 Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeStatusItem> TradeStatusItem </tradeStatusItem> [1..*]
   'A collection of data values describing the state of the given trade.'

   <party> Party </party> [1..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the
   term of the trade, but may include other parties involved in, or incidental to, the trade, such as
   parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML
   roles are defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeStatus">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="tradeStatusItem" type=" TradeStatusItem " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeStatusItem

[Table of contents]

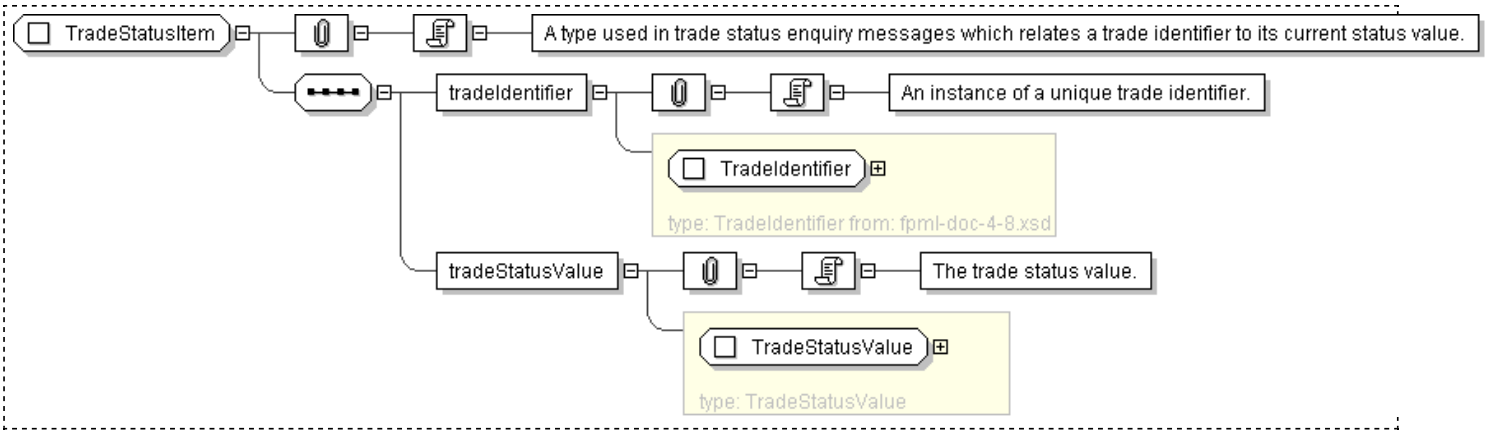
Super-types:	None
Sub-types:	None

Name	TradeStatusItem
Used by (from the same schema document)	Complex Type TradeStatus
Abstract	no
Documentation	A type used in trade status enquiry messages which relates a trade identifier to its current status value.

XML Instance Representation

```
<...>
  <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
  'An instance of a unique trade identifier.'

  <tradeStatusValue> TradeStatusValue </tradeStatusValue> [1]
  'The trade status value.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeStatusItem">
  <xsd:sequence>
    <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
    <xsd:element name="tradeStatusValue" type=" TradeStatusValue "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeStatusValue

[Table of contents]

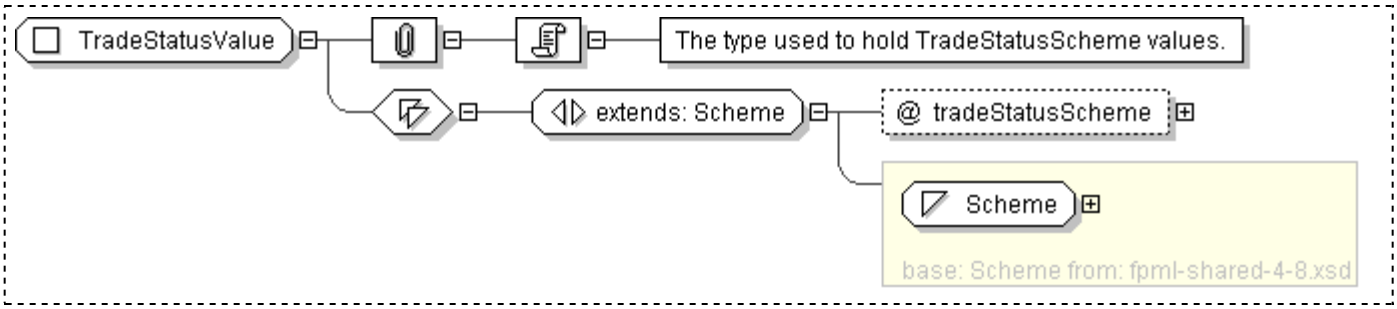
Super-types:	Scheme < TradeStatusValue (by extension)
Sub-types:	None

Name	TradeStatusValue
Used by (from the same schema document)	Complex Type TradeStatusItem
Abstract	no
Documentation	The type used to hold TradeStatusScheme values.

XML Instance Representation

```
<...
 tradeStatusScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeStatusValue">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="tradeStatusScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: Asian
  - Complex Type: AveragingObservationList
  - Complex Type: AveragingPeriod
  - Complex Type: AveragingSchedule
  - Complex Type: Barrier
  - Complex Type: CalendarSpread
  - Complex Type: ClassifiedPayment
  - Complex Type: Composite
  - Complex Type: CreditEventNotice
  - Complex Type: CreditEvents
  - Complex Type: CreditEventsReference
  - Complex Type: FailureToPay
  - Complex Type: FeaturePayment
  - Complex Type: FrequencyType
  - Complex Type: FxFeature
  - Complex Type: GracePeriodExtension
  - Complex Type: Knock
  - Complex Type: MarketDisruption
  - Complex Type: NotifyingParty
  - Complex Type: OptionBase
  - Complex Type: OptionBaseExtended
  - Complex Type: OptionFeature
  - Complex Type: OptionNumericStrike
  - Complex Type: OptionStrike
  - Complex Type: PassThrough
  - Complex Type: PassThroughItem
  - Complex Type: Premium
  - Complex Type: PubliclyAvailableInformation
  - Complex Type: Quanto
  - Complex Type: Restructuring
  - Complex Type: RestructuringType
  - Complex Type: StrategyFeature
  - Complex Type: StrikeSpread
  - Complex Type: Trigger
  - Complex Type: TriggerEvent
  - Complex Type: WeightedAveragingObservation
  - Model Group: OptionBaseFeature.model
  - Model Group: OptionDenomination.model
  - Model Group: OptionFeature.model
  - Model Group: OptionSettlement.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute	• Global element and attribute declarations belong to this schema's

Namespaces	target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-asset-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-asset-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
```

```
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by

specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: OptionBaseFeature.model

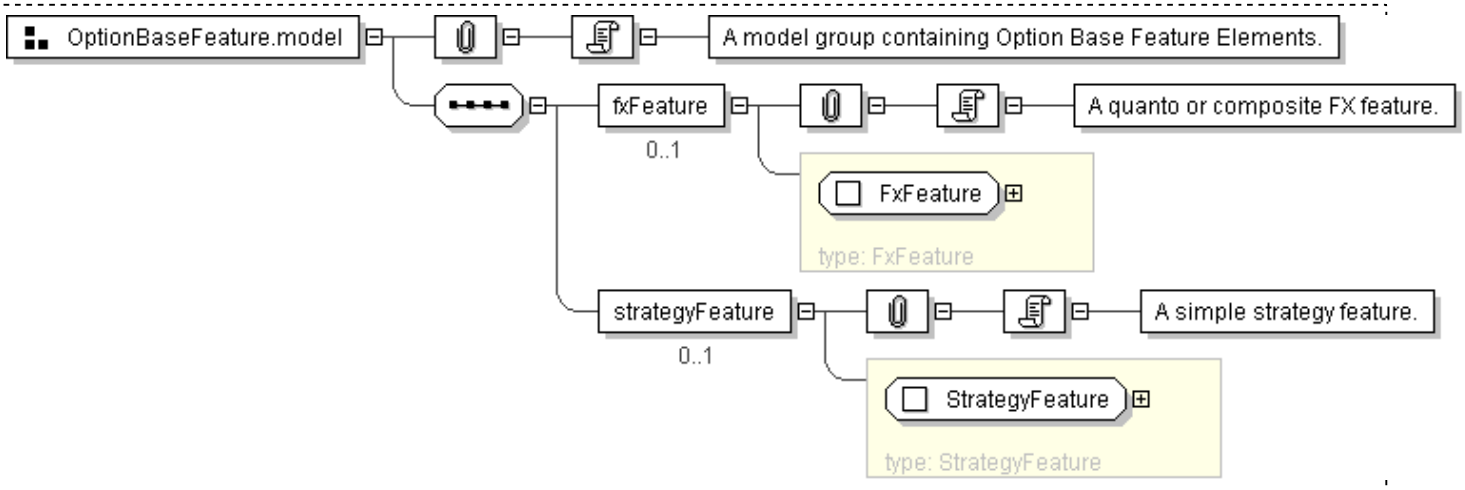
[Table of contents]

Name	OptionBaseFeature.model
Used by (from the same schema document)	Complex Type OptionFeature
Documentation	A model group containing Option Base Feature Elements.

XML Instance Representation

```
<fxFeature> FxFeature </fxFeature> [0..1]
'A quanto or composite FX feature.'

<strategyFeature> StrategyFeature </strategyFeature> [0..1]
'A simple strategy feature.'
```

Diagram



Schema Component Representation

```
<xsd:group name="OptionBaseFeature.model">
  <xsd:sequence>
    <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
    <xsd:element name="strategyFeature" type=" StrategyFeature " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: OptionDenomination.model

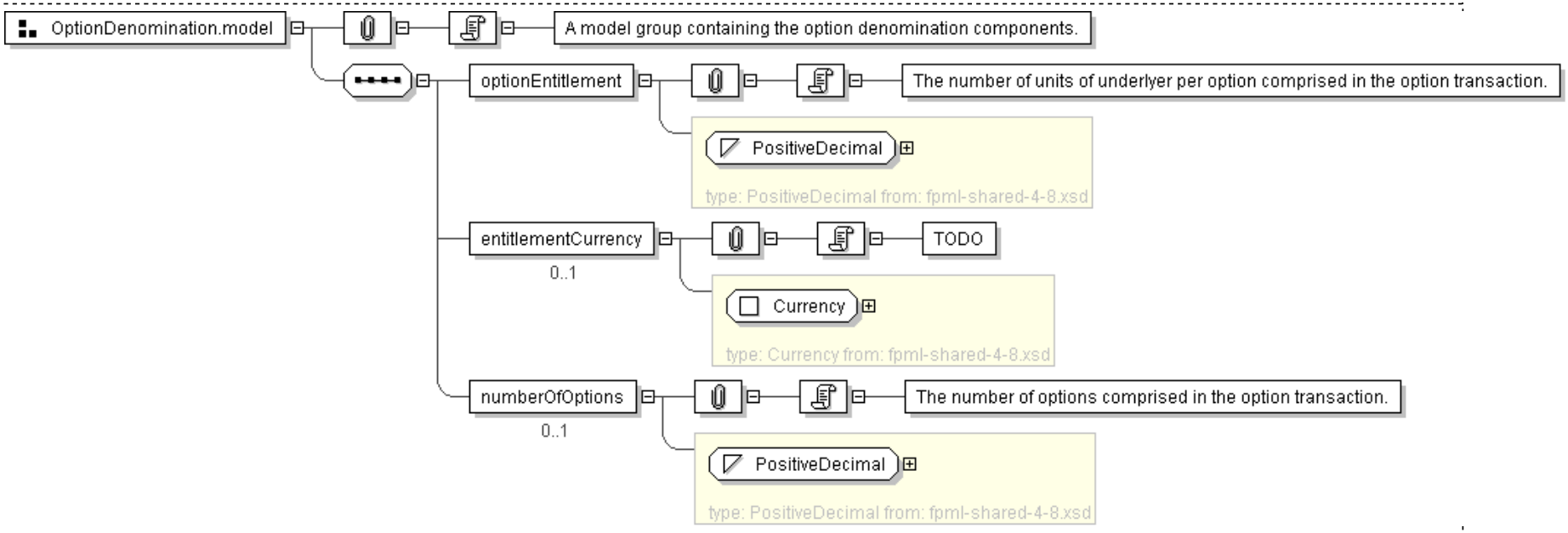
[Table of contents]

Name	OptionDenomination.model
Used by (from the same schema document)	Complex Type OptionBaseExtended
Documentation	A model group containing the option denomination components.

XML Instance Representation

```
<optionEntitlement> PositiveDecimal </optionEntitlement> [1]
 'The number of units of underlyer per option comprised in the option transaction.'

<entitlementCurrency> Currency </entitlementCurrency> [0..1]
 'TODO'

<numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
 'The number of options comprised in the option transaction.'
```

Diagram



Schema Component Representation

```
<xsd:group name="OptionDenomination.model">
  <xsd:sequence>
    <xsd:element name="optionEntitlement" type=" PositiveDecimal "/>
    <xsd:element name="entitlementCurrency" type=" Currency " minOccurs="0"/>
```

```
    <xsd:element name="numberOfOptions" type=" PositiveDecimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: OptionFeature.model

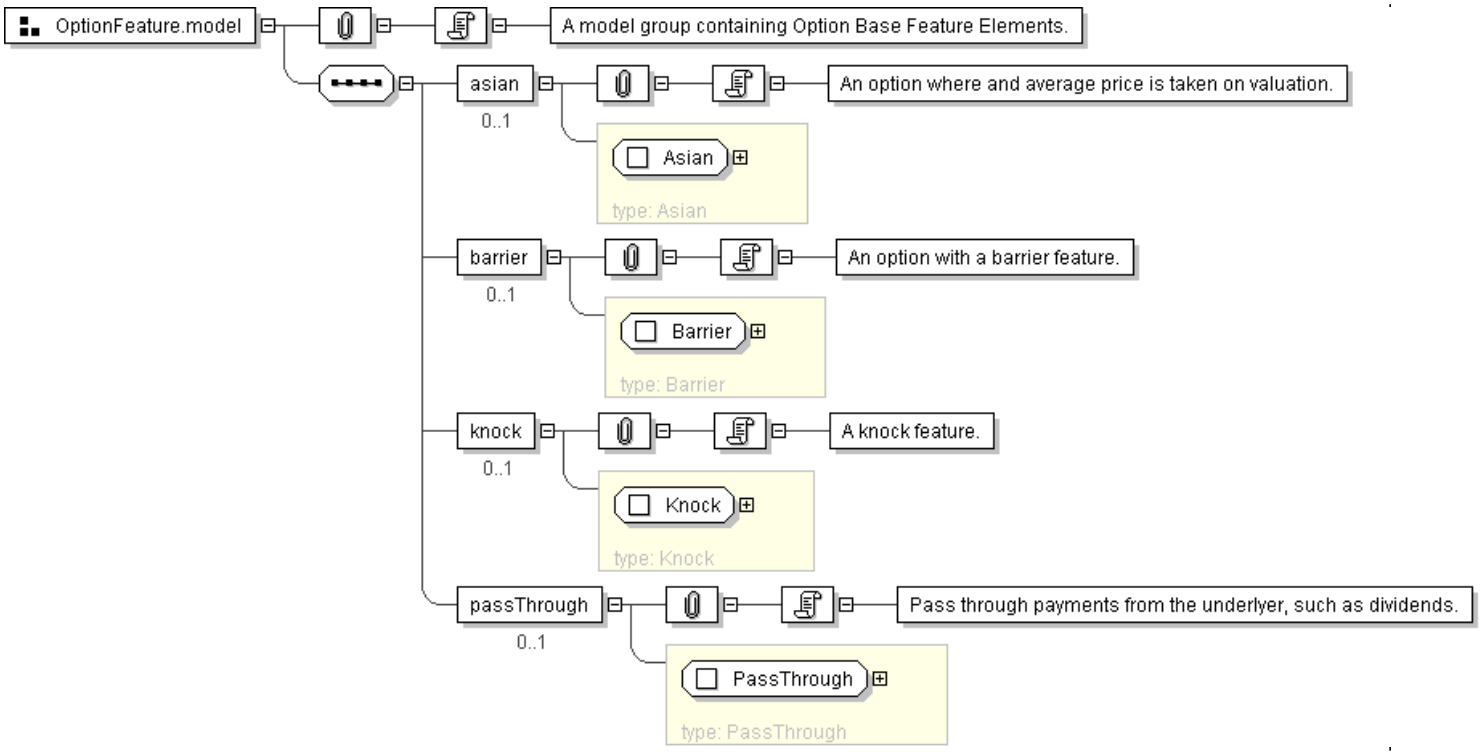
[Table of contents]

Name	OptionFeature.model
Used by (from the same schema document)	Complex Type OptionFeature
Documentation	A model group containing Option Base Feature Elements.

XML Instance Representation

```
<asian> Asian </asian> [0..1]
'An option where and average price is taken on valuation.'

<barrier> Barrier </barrier> [0..1]
'An option with a barrier feature.'

<knock> Knock </knock> [0..1]
'A knock feature.'

<passThrough> PassThrough </passThrough> [0..1]
'Pass through payments from the underlyer, such as dividends.'
```

Diagram



Schema Component Representation

```
<xsd:group name="OptionFeature.model">
  <xsd:sequence>
    <xsd:element name="asian" type=" Asian " minOccurs="0"/>
    <xsd:element name="barrier" type=" Barrier " minOccurs="0"/>
    <xsd:element name="knock" type=" Knock " minOccurs="0"/>
    <xsd:element name="passThrough" type=" PassThrough " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: OptionSettlement.model

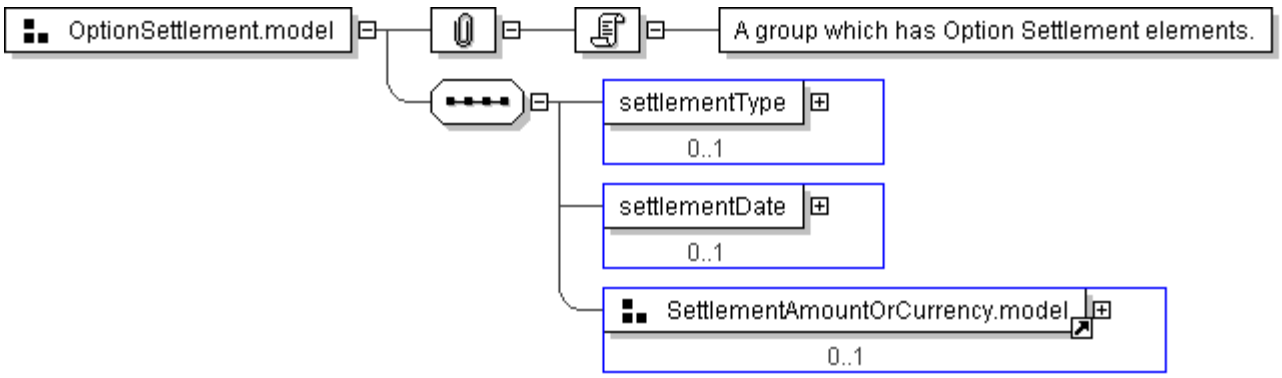
[Table of contents]

Name	OptionSettlement.model
Used by (from the same schema document)	Complex Type OptionBaseExtended
Documentation	A group which has Option Settlement elements.

XML Instance Representation

```
<settlementType> SettlementTypeEnum </settlementType> [0..1]
<settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
Start Group: SettlementAmountOrCurrency.model [0..1]
   Start Choice [1]
      <settlementAmount> Money </settlementAmount> [1]
      'Settlement Amount'

      <settlementCurrency> Currency </settlementCurrency> [1]
      'Settlement Currency for use where the Settlement Amount cannot be known in
      advance'

   End Choice
End Group: SettlementAmountOrCurrency.model
```

Diagram



Schema Component Representation

```
<xsd:group name="OptionSettlement.model">
   <xsd:sequence>
      <xsd:element name="settlementType" type=" SettlementTypeEnum " minOccurs="0"/>
      <xsd:element name="settlementDate" type=" AdjustableOrRelativeDate "
      minOccurs="0"/>
      <xsd:group ref=" SettlementAmountOrCurrency.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Asian

[Table of contents]

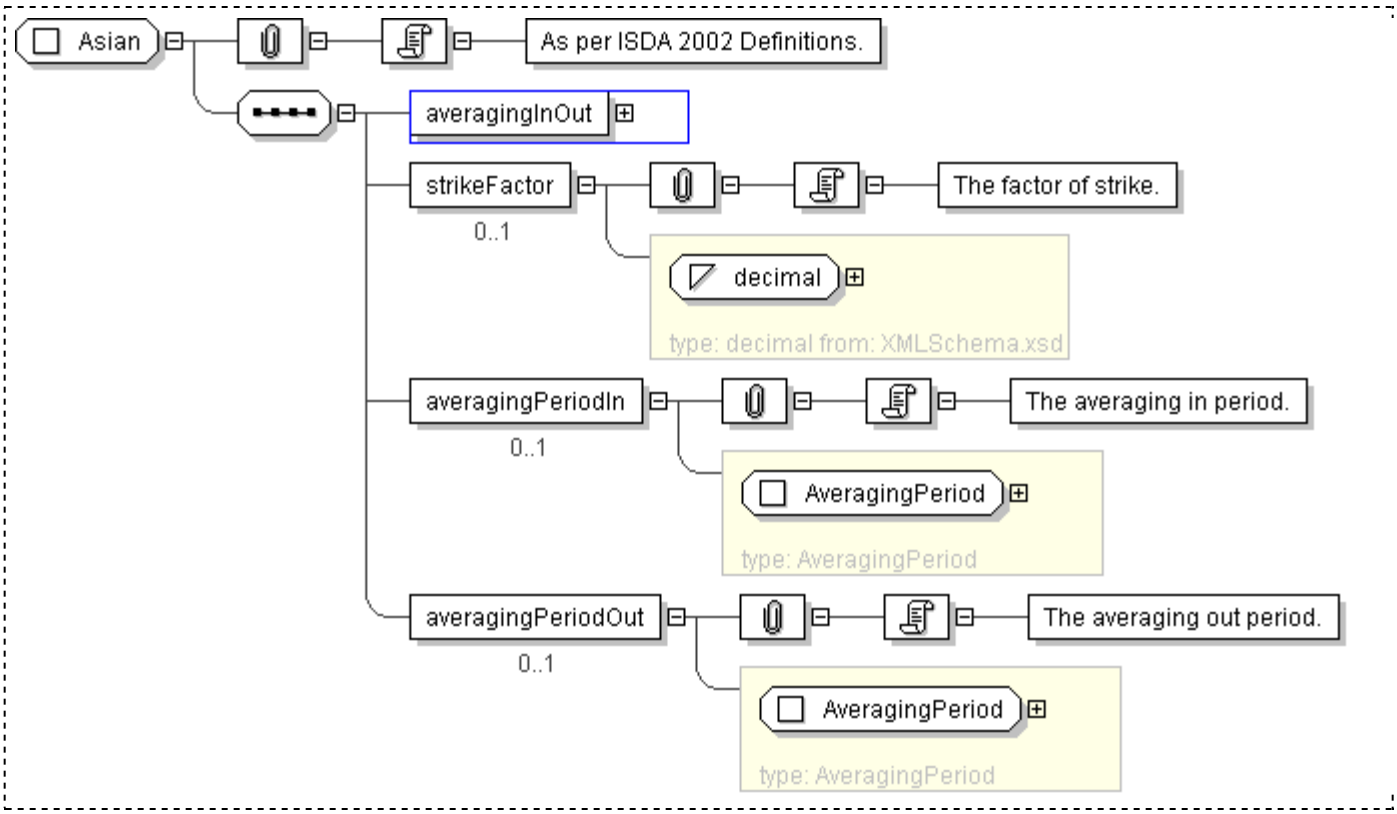
Super-types:	None
Sub-types:	None

Name	Asian
Used by (from the same schema document)	Model Group OptionFeature.model
Abstract	no
Documentation	As per ISDA 2002 Definitions.

XML Instance Representation

```
<...>
   <averagingInOut> AveragingInOutEnum </averagingInOut> [1]
   <strikeFactor> xsd:decimal </strikeFactor> [0..1]
    'The factor of strike.'

   <averagingPeriodIn> AveragingPeriod </averagingPeriodIn> [0..1]
    'The averaging in period.'

   <averagingPeriodOut> AveragingPeriod </averagingPeriodOut> [0..1]
    'The averaging out period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Asian">
   <xsd:sequence>
      <xsd:element name="averagingInOut" type=" AveragingInOutEnum "/>
      <xsd:element name="strikeFactor" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="averagingPeriodIn" type=" AveragingPeriod " minOccurs="0"/>
      <xsd:element name="averagingPeriodOut" type=" AveragingPeriod " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AveragingObservationList

[Table of contents]

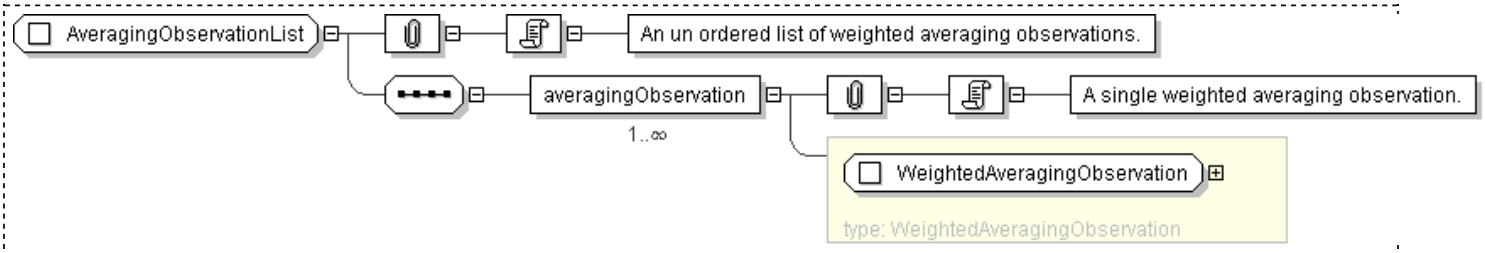
Super-types:	None
Sub-types:	None

Name	AveragingObservationList
Used by (from the same schema document)	Complex Type AveragingPeriod
Abstract	no
Documentation	An un ordered list of weighted averaging observations.

XML Instance Representation

```
<...>
   <averagingObservation> WeightedAveragingObservation </averagingObservation> [1..*]
   'A single weighted averaging observation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AveragingObservationList">
   <xsd:sequence>
      <xsd:element name="averagingObservation" type=" WeightedAveragingObservation "
      maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AveragingPeriod

[Table of contents]

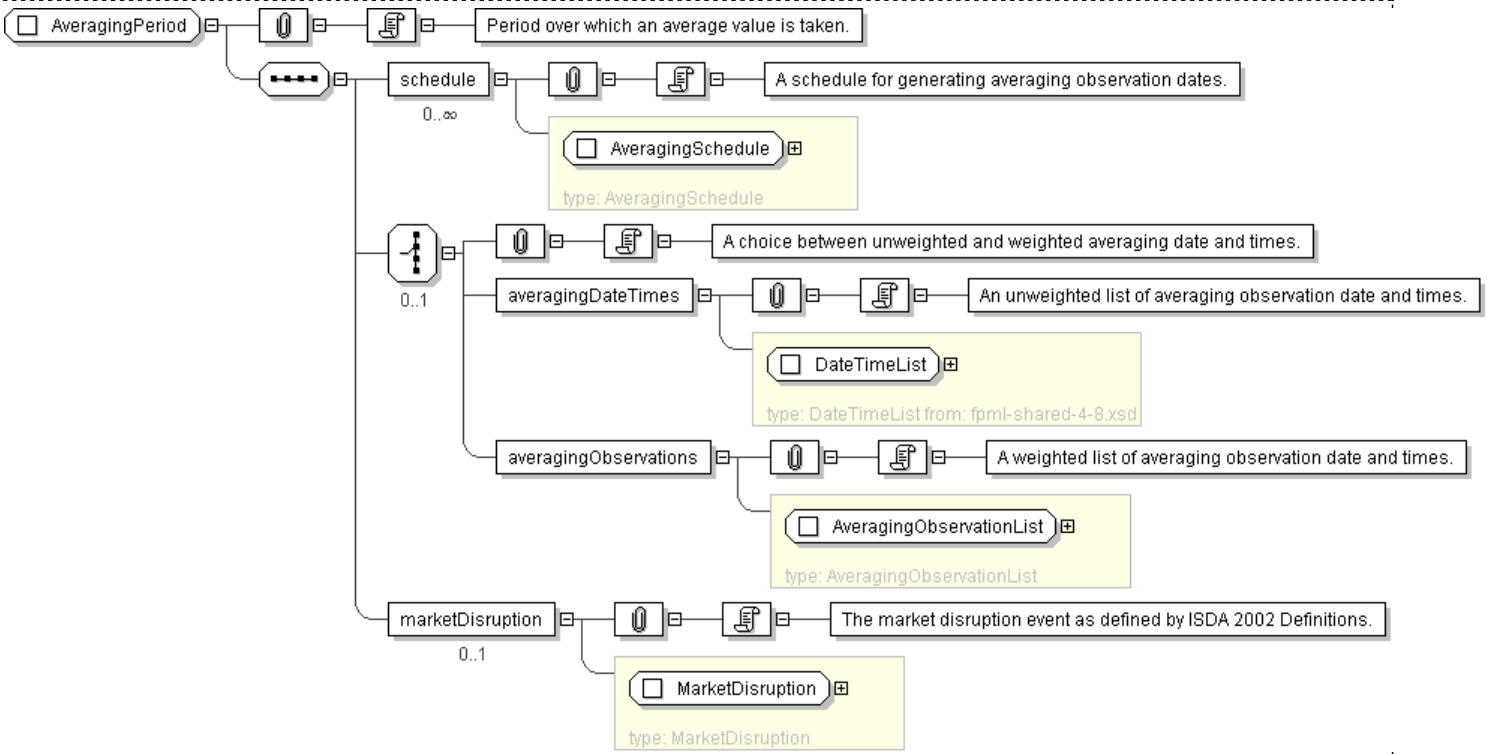
Super-types:	None
Sub-types:	None

Name	AveragingPeriod
Used by (from the same schema document)	Complex Type Asian , Complex Type Asian
Abstract	no
Documentation	Period over which an average value is taken.

XML Instance Representation

```
<...>
   <schedule> AveragingSchedule </schedule> [0..*]
   'A schedule for generating averaging observation dates.'

   Start Choice [0..1]
'A choice between unweighted and weighted averaging date and times.'


      <averagingDateTimes> DateTimeList </averagingDateTimes> [1]
      'An unweighted list of averaging observation date and times.'

      <averagingObservations> AveragingObservationList </averagingObservations> [1]
      'A weighted list of averaging observation date and times.'

   End Choice
   <marketDisruption> MarketDisruption </marketDisruption> [0..1]
   'The market disruption event as defined by ISDA 2002 Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AveragingPeriod">
   <xsd:sequence>
      <xsd:element name="schedule" type=" AveragingSchedule " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:choice minOccurs="0">
         <xsd:element name="averagingDateTimes" type=" DateTimeList "/>
         <xsd:element name="averagingObservations" type=" AveragingObservationList "/>
      </xsd:choice>
```

```
      <xsd:element name="marketDisruption" type=" MarketDisruption " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AveragingSchedule

[Table of contents]

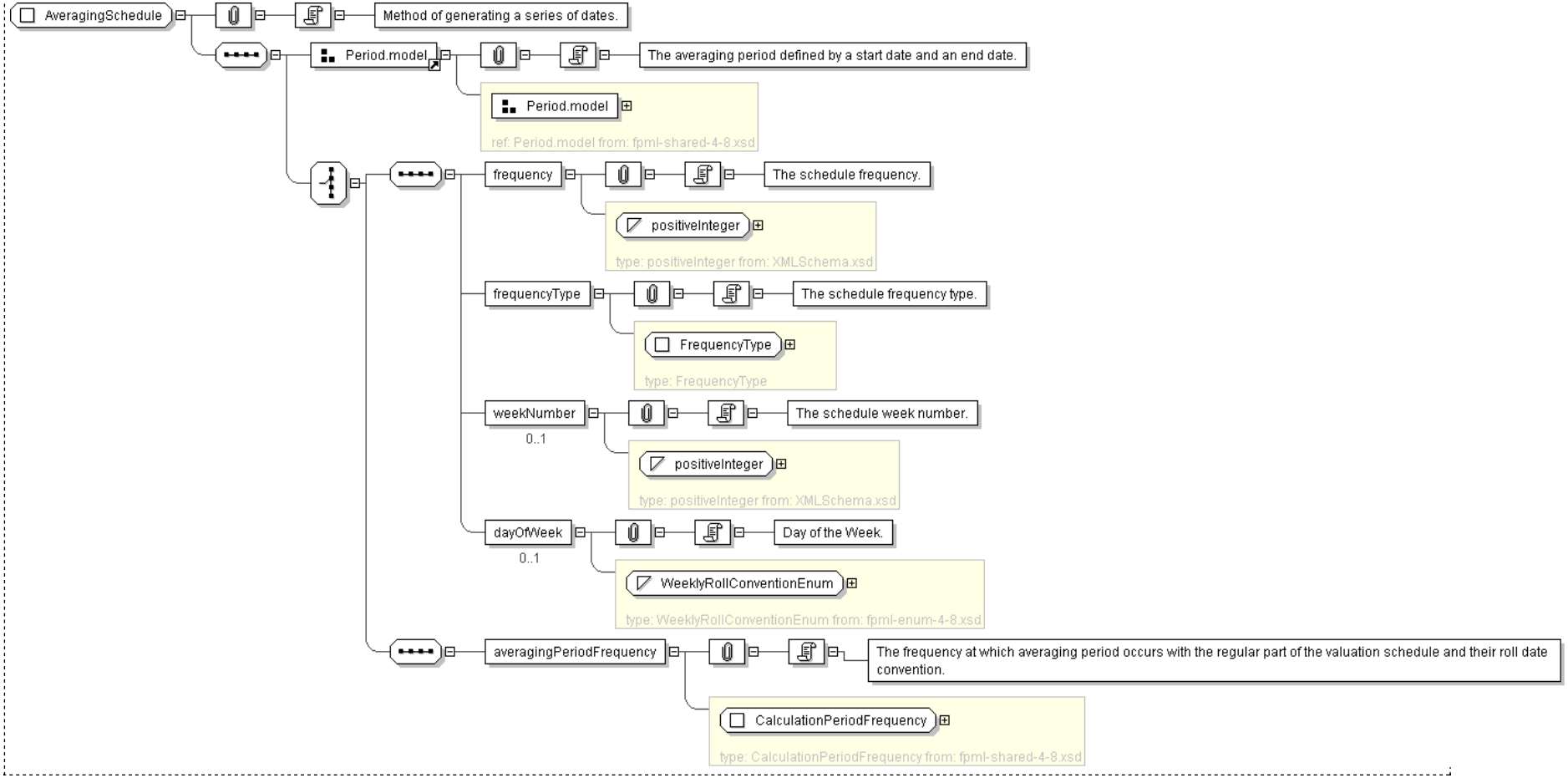
Super-types:	None
Sub-types:	None

Name	AveragingSchedule
Used by (from the same schema document)	Complex Type AveragingPeriod , Complex Type TriggerEvent
Abstract	no
Documentation	Method of generating a series of dates.

XML Instance Representation

```
<...>
   <startDate> xsd:date </startDate> [1]
    'Date on which this period begins.'

   <endDate> xsd:date </endDate> [1]
    'Date on which this period ends.'

   Start Choice [1]
      <frequency> xsd:positiveInteger </frequency> [1]
       'The schedule frequency.'

      <frequencyType> FrequencyType </frequencyType> [1]
       'The schedule frequency type.'

      <weekNumber> xsd:positiveInteger </weekNumber> [0..1]
       'The schedule week number.'

      <dayOfWeek> WeeklyRollConventionEnum </dayOfWeek> [0..1]
       'Day of the Week.'

      <averagingPeriodFrequency> CalculationPeriodFrequency </averagingPeriodFrequency> [1]
       'The frequency at which averaging period occurs with the regular part of the valuation schedule and their roll date convention.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AveragingSchedule">
  <xsd:sequence>
    <xsd:group ref=" Period.model "/>
    <xsd:choice>
      <xsd:sequence>
        <xsd:element name="frequency" type=" xsd:positiveInteger "/>
        <xsd:element name="frequencyType" type=" FrequencyType "/>
        <xsd:element name="weekNumber" type=" xsd:positiveInteger " minOccurs="0"/>
        <xsd:element name="dayOfWeek" type=" WeeklyRollConventionEnum " minOccurs="0"/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="averagingPeriodFrequency" type=" CalculationPeriodFrequency "/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Barrier

[Table of contents]

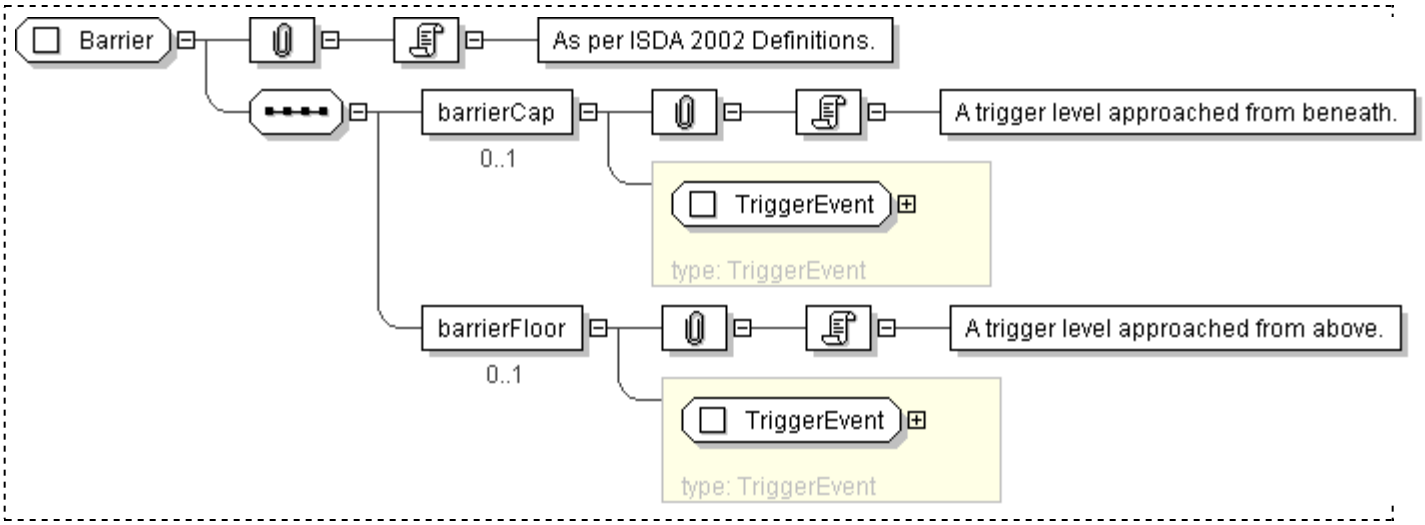
Super-types:	None
Sub-types:	None

Name	Barrier
Used by (from the same schema document)	Model Group OptionFeature.model
Abstract	no
Documentation	As per ISDA 2002 Definitions.

XML Instance Representation

```
<...>
   <barrierCap> TriggerEvent </barrierCap> [0..1]
    'A trigger level approached from beneath.'

   <barrierFloor> TriggerEvent </barrierFloor> [0..1]
    'A trigger level approached from above.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Barrier">
   <xsd:sequence>
      <xsd:element name="barrierCap" type=" TriggerEvent " minOccurs="0"/>
      <xsd:element name="barrierFloor" type=" TriggerEvent " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalendarSpread

[Table of contents]

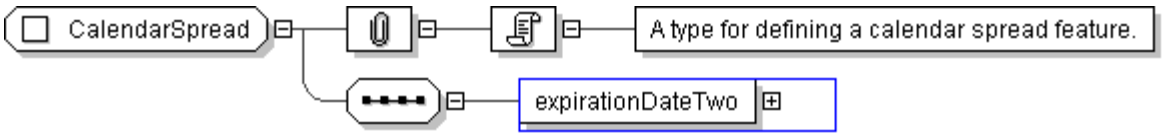
Super-types:	None
Sub-types:	None

Name	CalendarSpread
Used by (from the same schema document)	Complex Type StrategyFeature
Abstract	no
Documentation	A type for defining a calendar spread feature.

XML Instance Representation

```
<...>
   <expirationDateTwo> AdjustableOrRelativeDate </expirationDateTwo> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalendarSpread">
   <xsd:sequence>
      <xsd:element name="expirationDateTwo" type=" AdjustableOrRelativeDate "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ClassifiedPayment

[Table of contents]

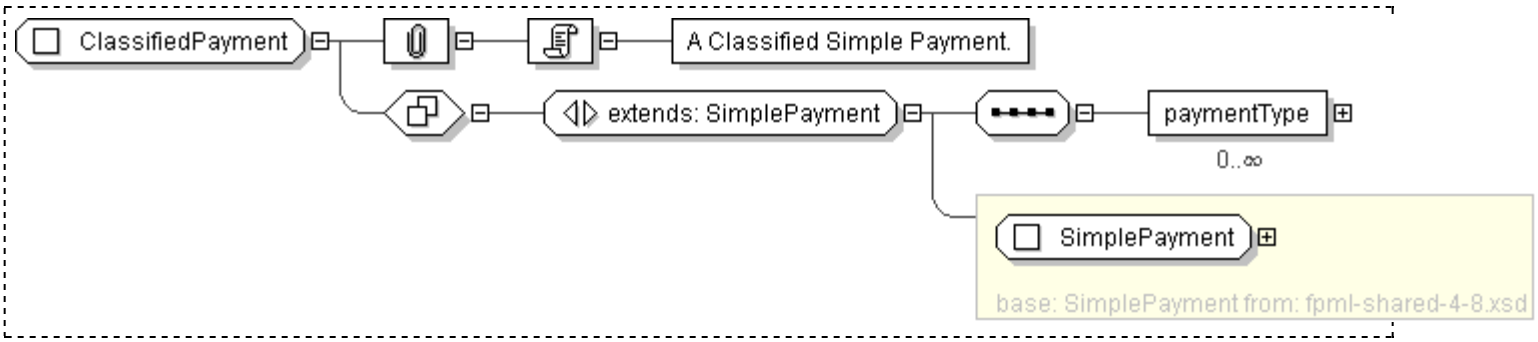
Super-types:	SimplePayment < ClassifiedPayment (by extension)
Sub-types:	None

Name	ClassifiedPayment
Abstract	no
Documentation	A Classified Simple Payment.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentAmount> Money </paymentAmount> [1]
   <paymentDate> AdjustableOrRelativeAndAdjustedDate </paymentDate> [1]
   'The payment date. This date is subject to adjustment in accordance with any
   applicable business day convention.'

   <paymentType> PaymentType </paymentType> [0..*]
   'Payment classification.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ClassifiedPayment">
   <xsd:complexContent>
      <xsd:extension base=" SimplePayment ">
         <xsd:sequence>
            <xsd:element name="paymentType" type=" PaymentType " minOccurs="0"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Composite

[Table of contents]

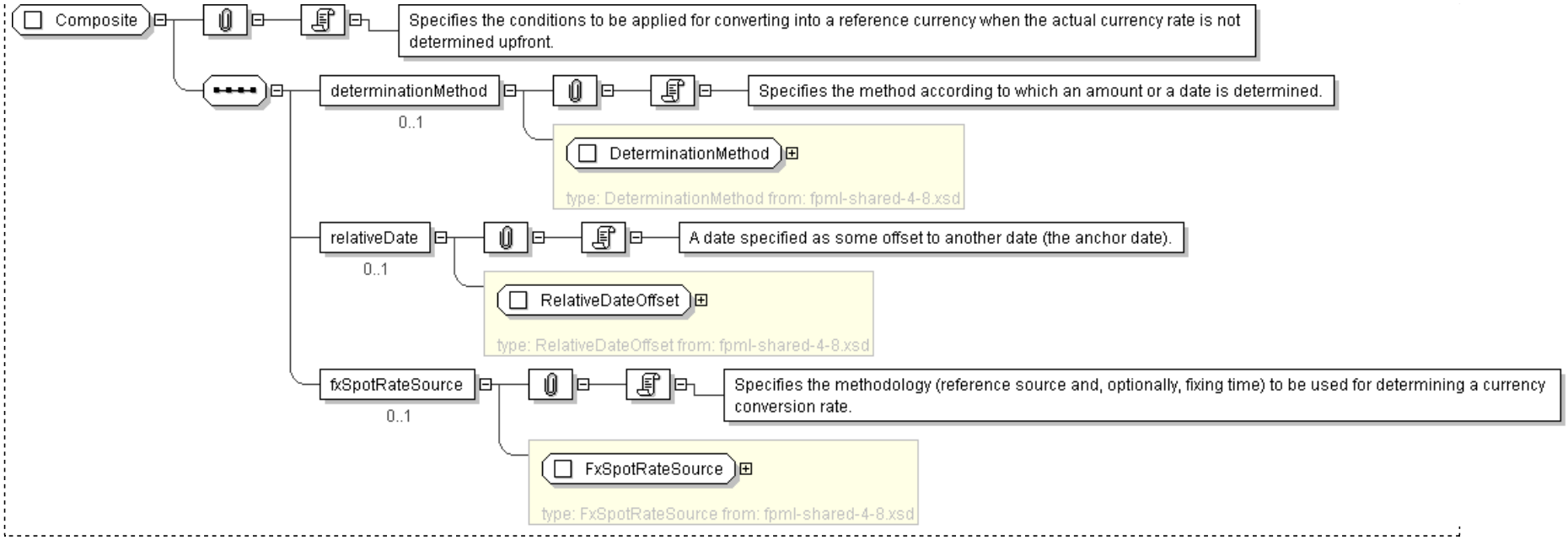
Super-types:	None
Sub-types:	None

Name	Composite
Used by (from the same schema document)	Complex Type FxFeature , Complex Type FxFeature
Abstract	no
Documentation	Specifies the conditions to be applied for converting into a reference currency when the actual currency rate is not determined upfront.

XML Instance Representation

```
<...>
   <determinationMethod> DeterminationMethod </determinationMethod> [0..1]
   'Specifies the method according to which an amount or a date is determined.'

   <relativeDate> RelativeDateOffset </relativeDate> [0..1]
   'A date specified as some offset to another date (the anchor date).'

   <fxSpotRateSource> FxSpotRateSource </fxSpotRateSource> [0..1]
   'Specifies the methodology (reference source and, optionally, fixing time) to be used for determining a currency conversion
   rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Composite">
  <xsd:sequence>
    <xsd:element name="determinationMethod" type=" DeterminationMethod " minOccurs="0"/>
    <xsd:element name="relativeDate" type=" RelativeDateOffset " minOccurs="0"/>
    <xsd:element name="fxSpotRateSource" type=" FxSpotRateSource " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEventNotice

[Table of contents]

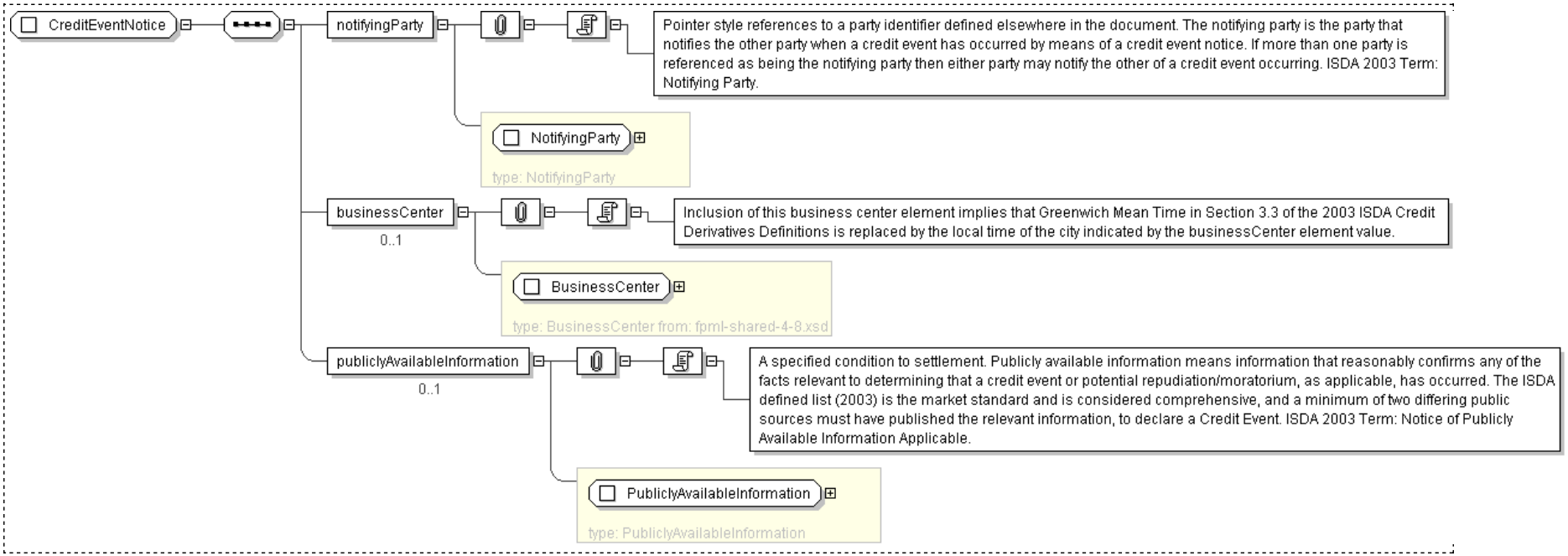
Super-types:	None
Sub-types:	None

Name	CreditEventNotice
Used by (from the same schema document)	Complex Type CreditEvents
Abstract	no

XML Instance Representation

```
<...>
  <notifyingParty> NotifyingParty </notifyingParty> [1]
    'Pointer style references to a party identifier defined elsewhere in the document. The notifying party is the party that notifies the other
    party when a credit event has occurred by means of a credit event notice. If more than one party is referenced as being the notifying party
    then either party may notify the other of a credit event occurring. ISDA 2003 Term: Notifying Party.'

  <businessCenter> BusinessCenter </businessCenter> [0..1]
    'Inclusion of this business center element implies that Greenwich Mean Time in Section 3.3 of the 2003 ISDA Credit Derivatives Definitions is
    replaced by the local time of the city indicated by the businessCenter element value.'

  <publiclyAvailableInformation> PubliclyAvailableInformation </publiclyAvailableInformation> [0..1]
    'A specified condition to settlement. Publicly available information means information that reasonably confirms any of the facts relevant to
    determining that a credit event or potential repudiation/moratorium, as applicable, has occurred. The ISDA defined list (2003) is the market
    standard and is considered comprehensive, and a minimum of two differing public sources must have published the relevant information, to declare
    a Credit Event. ISDA 2003 Term: Notice of Publicly Available Information Applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditEventNotice">
   <xsd:sequence>
      <xsd:element name="notifyingParty" type=" NotifyingParty "/>
      <xsd:element name="businessCenter" type=" BusinessCenter " minOccurs="0"/>
      <xsd:element name="publiclyAvailableInformation" type=" PubliclyAvailableInformation " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEvents

[Table of contents]

Super-types:	None
Sub-types:	None

Name	CreditEvents
Used by (from the same schema document)	Complex Type Trigger
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <bankruptcy> Empty </bankruptcy> [0..1]
   'A credit event. The reference entity has been dissolved or has become insolvent. It also covers events that may be a precursor to insolvency
   such as instigation of bankruptcy or insolvency proceedings. Sovereign trades are not subject to Bankruptcy as \"technically\" a Sovereign
   cannot become bankrupt. ISDA 2003 Term: Bankruptcy.'

   <failureToPay> FailureToPay </failureToPay> [0..1]
   'A credit event. This credit event triggers, after the expiration of any applicable grace period, if the reference entity fails to make due
   payments in an aggregrate amount of not less than the payment requirement on one or more obligations (e.g. a missed coupon payment). ISDA 2003
   Term: Failure to Pay.'

   <failureToPayPrincipal> Empty </failureToPayPrincipal> [0..1]
   'A credit event. Corresponds to the failure by the Reference Entity to pay an expected principal amount or the payment of an actual principal
   amount that is less than the expected principal amount. ISDA 2003 Term: Failure to Pay Principal.'

   <failureToPayInterest> Empty </failureToPayInterest> [0..1]
   'A credit event. Corresponds to the failure by the Reference Entity to pay an expected interest amount or the payment of an actual interest
   amount that is less than the expected interest amount. ISDA 2003 Term: Failure to Pay Interest.'

   <obligationDefault> Empty </obligationDefault> [0..1]
   'A credit event. One or more of the obligations have become capable of being declared due and payable before they would otherwise have been
   due and payable as a result of, or on the basis of, the occurrence of a default, event of default or other similar condition or event other
   than failure to pay. ISDA 2003 Term: Obligation Default.'

   <obligationAcceleration> Empty </obligationAcceleration> [0..1]
   'A credit event. One or more of the obligations have been declared due and payable before they would otherwise have been due and payable as a
   result of, or on the basis of, the occurrence of a default, event of default or other similar condition or event other than failure to pay
   (preferred by the market over Obligation Default, because more definitive and encompasses the definition of Obligation Default - this is more
   favorable to the Seller). Subject to the default requirement amount. ISDA 2003 Term: Obligation Acceleration.'

   <repudiationMoratorium> Empty </repudiationMoratorium> [0..1]
   'A credit event. The reference entity, or a governmental authority, either refuses to recognise or challenges the validity of one or more
   obligations of the reference entity, or imposes a moratorium thereby postponing payments on one or more of the obligations of the reference
   entity. Subject to the default requirement amount. ISDA 2003 Term: Repudiation/Moratorium.'

   <restructuring> Restructuring </restructuring> [0..1]
   'A credit event. A restructuring is an event that materially impacts the reference entity\'s obligations, such as an interest rate reduction,
   principal reduction, deferral of interest or principal, change in priority ranking, or change in currency or composition of payment. ISDA 2003
   Term: Restructuring.'

   <distressedRatingsDowngrade> Empty </distressedRatingsDowngrade> [0..1]
   'A credit event. Results from the fact that the rating of the reference obligation is downgraded to a distressed rating level. From a usage
   standpoint, this credit event is typically not applicable in case of RMBS trades.'

   <maturityExtension> Empty </maturityExtension> [0..1]
   'A credit event. Results from the fact that the underlier fails to make principal payments as expected.'

   <writedown> Empty </writedown> [0..1]
   'A credit event. Results from the fact that the underlier writes down its outstanding principal amount.'

   <impliedWritedown> Empty </impliedWritedown> [0..1]
   'A credit event. Results from the fact that losses occur to the underlying instruments that do not result in reductions of the outstanding
   principal of the reference obligation.'

   <defaultRequirement> Money </defaultRequirement> [0..1]
```

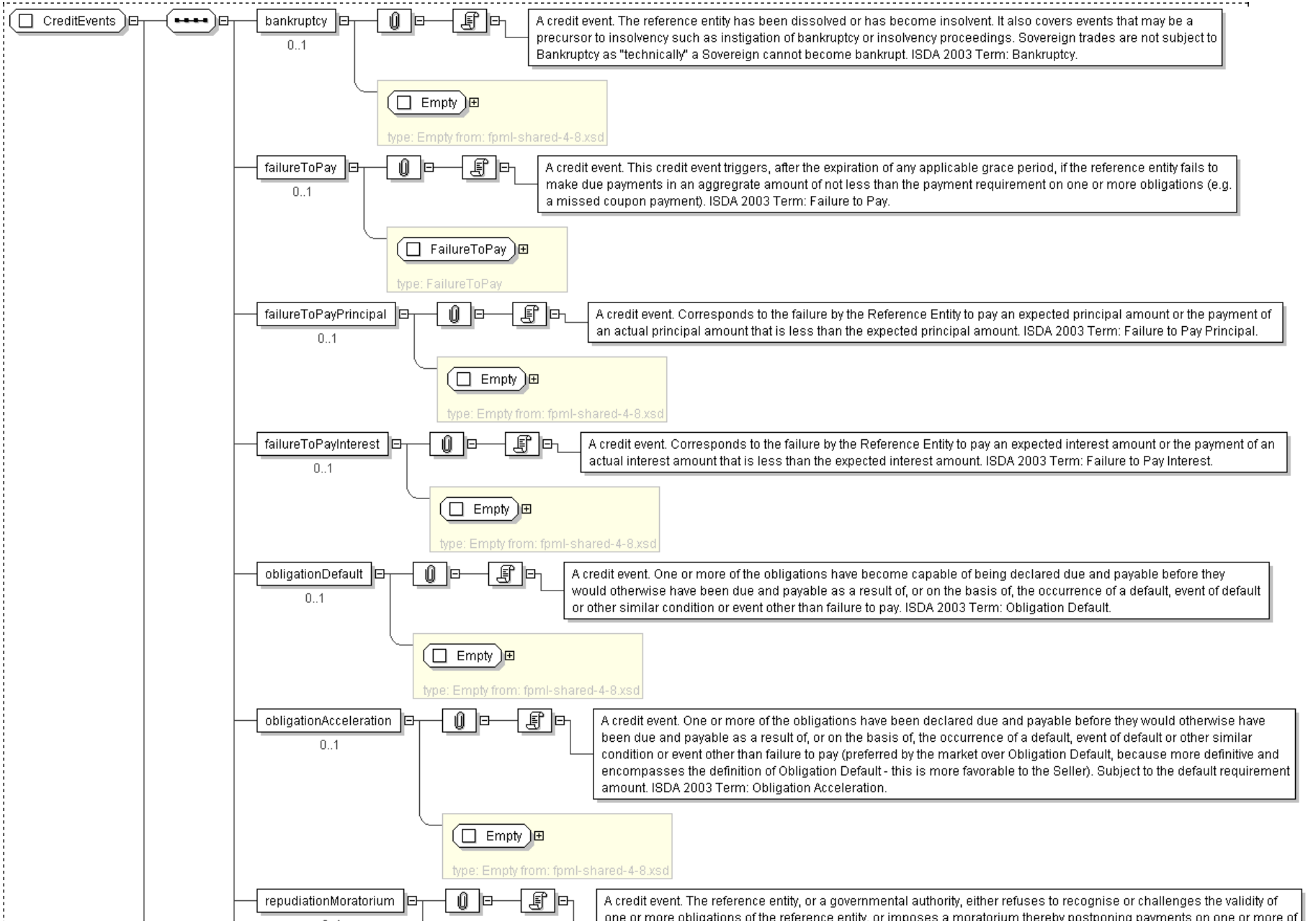
'In relation to certain credit events, serves as a threshold for Obligation Acceleration, Obligation Default, Repudiation/Moratorium and Restructuring. Market standard is USD 10,000,000 (JPY 1,000,000,000 for all Japanese Yen trades). This is applied on an aggregate or total basis across all Obligations of the Reference Entity. Used to prevent technical/operational errors from triggering credit events. ISDA 2003 Term: Default Requirement.'

```
<creditEventNotice> CreditEventNotice </creditEventNotice> [0..1]
```

'A specified condition to settlement. An irrevocable written or verbal notice that describes a credit event that has occurred. The notice is sent from the notifying party (either the buyer or the seller) to the counterparty. It provides information relevant to determining that a credit event has occurred. This is typically accompanied by Publicly Available Information. ISDA 2003 Term: Credit Event Notice.'

```
</...>
```

Diagram



0..1

the obligations of the reference entity. Subject to the default requirement amount. ISDA 2003 Term: Repudiation/Moratorium.

Empty
type: Empty from: fpml-shared-4-8.xsd

restructuring
0..1

A credit event. A restructuring is an event that materially impacts the reference entity's obligations, such as an interest rate reduction, principal reduction, deferral of interest or principal, change in priority ranking, or change in currency or composition of payment. ISDA 2003 Term: Restructuring.

Restructuring
type: Restructuring

distressedRatingsDowngrade
0..1

A credit event. Results from the fact that the rating of the reference obligation is downgraded to a distressed rating level. From a usage standpoint, this credit event is typically not applicable in case of RMBS trades.

Empty
type: Empty from: fpml-shared-4-8.xsd

maturityExtension
0..1

A credit event. Results from the fact that the underlier fails to make principal payments as expected.

Empty
type: Empty from: fpml-shared-4-8.xsd

writedown
0..1

A credit event. Results from the fact that the underlier writes down its outstanding principal amount.

Empty
type: Empty from: fpml-shared-4-8.xsd

impliedWritedown
0..1

A credit event. Results from the fact that losses occur to the underlying instruments that do not result in reductions of the outstanding principal of the reference obligation.

Empty
type: Empty from: fpml-shared-4-8.xsd

defaultRequirement
0..1

In relation to certain credit events, serves as a threshold for Obligation Acceleration, Obligation Default, Repudiation/Moratorium and Restructuring. Market standard is USD 10,000,000 (JPY 1,000,000,000 for all Japanese Yen trades). This is applied on an aggregate or total basis across all Obligations of the Reference Entity. Used to prevent technical/operational errors from triggering credit events. ISDA 2003 Term: Default Requirement.

Money
type: Money from: fpml-shared-4-8.xsd

creditEventNotice
0..1

A specified condition to settlement. An irrevocable written or verbal notice that describes a credit event that has occurred. The notice is sent from the notifying party (either the buyer or the seller) to the counterparty. It provides information relevant to determining that a credit event has occurred. This is typically accompanied by Publicly Available Information. ISDA 2003 Term: Credit Event Notice.

CreditEventNotice
type: CreditEventNotice

@ id

Schema Component Representation

```
<xsd:complexType name="CreditEvents">
   <xsd:sequence>
      <xsd:element name="bankruptcy" type=" Empty " minOccurs="0"/>
      <xsd:element name="failureToPay" type=" FailureToPay " minOccurs="0"/>
      <xsd:element name="failureToPayPrincipal" type=" Empty " minOccurs="0"/>
      <xsd:element name="failureToPayInterest" type=" Empty " minOccurs="0"/>
      <xsd:element name="obligationDefault" type=" Empty " minOccurs="0"/>
      <xsd:element name="obligationAcceleration" type=" Empty " minOccurs="0"/>
      <xsd:element name="repudiationMoratorium" type=" Empty " minOccurs="0"/>
      <xsd:element name="restructuring" type=" Restructuring " minOccurs="0"/>
      <xsd:element name="distressedRatingsDowngrade" type=" Empty " minOccurs="0"/>
      <xsd:element name="maturityExtension" type=" Empty " minOccurs="0"/>
      <xsd:element name="writedown" type=" Empty " minOccurs="0"/>
      <xsd:element name="impliedWritedown" type=" Empty " minOccurs="0"/>
      <xsd:element name="defaultRequirement" type=" Money " minOccurs="0"/>
      <xsd:element name="creditEventNotice" type=" CreditEventNotice " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditEventsReference

[Table of contents]

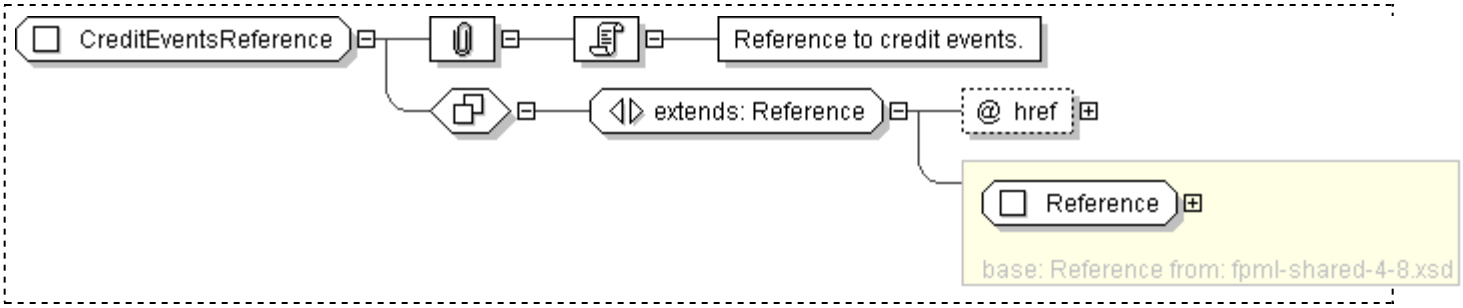
Super-types:	Reference < CreditEventsReference (by extension)
Sub-types:	None

Name	CreditEventsReference
Used by (from the same schema document)	Complex Type Trigger
Abstract	no
Documentation	Reference to credit events.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditEventsReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="CreditEvents"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FailureToPay

[Table of contents]

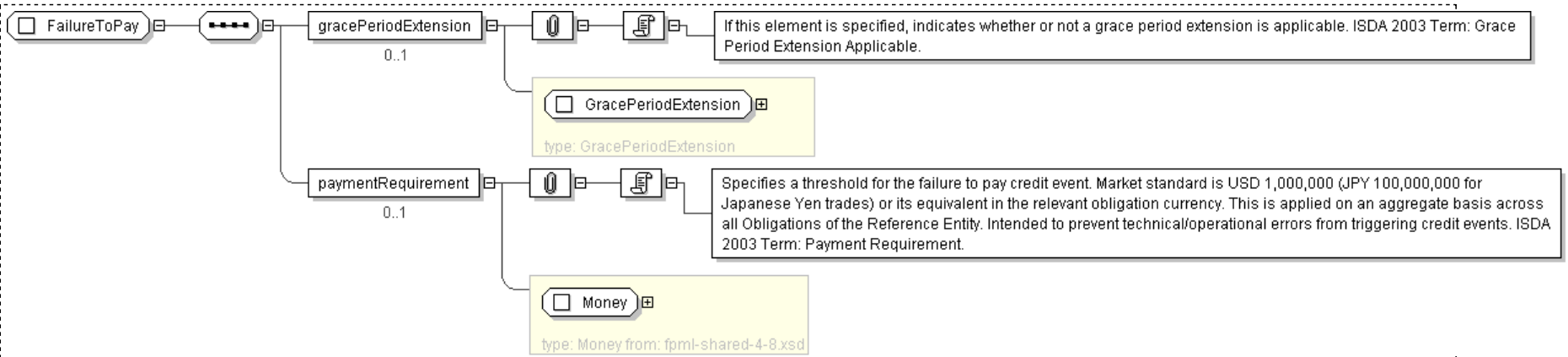
Super-types:	None
Sub-types:	None

Name	FailureToPay
Used by (from the same schema document)	Complex Type CreditEvents
Abstract	no

XML Instance Representation

```
<...>
   <gracePeriodExtension> GracePeriodExtension </gracePeriodExtension> [0..1]
   'If this element is specified, indicates whether or not a grace period extension is applicable. ISDA 2003 Term: Grace Period Extension
   Applicable.'

   <paymentRequirement> Money </paymentRequirement> [0..1]
   'Specifies a threshold for the failure to pay credit event. Market standard is USD 1,000,000 (JPY 100,000,000 for Japanese Yen trades) or
   its equivalent in the relevant obligation currency. This is applied on an aggregate basis across all Obligations of the Reference Entity.
   Intended to prevent technical/operational errors from triggering credit events. ISDA 2003 Term: Payment Requirement.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FailureToPay">
   <xsd:sequence>
      <xsd:element name="gracePeriodExtension" type=" GracePeriodExtension " minOccurs="0"/>
      <xsd:element name="paymentRequirement" type=" Money " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FeaturePayment

[Table of contents]

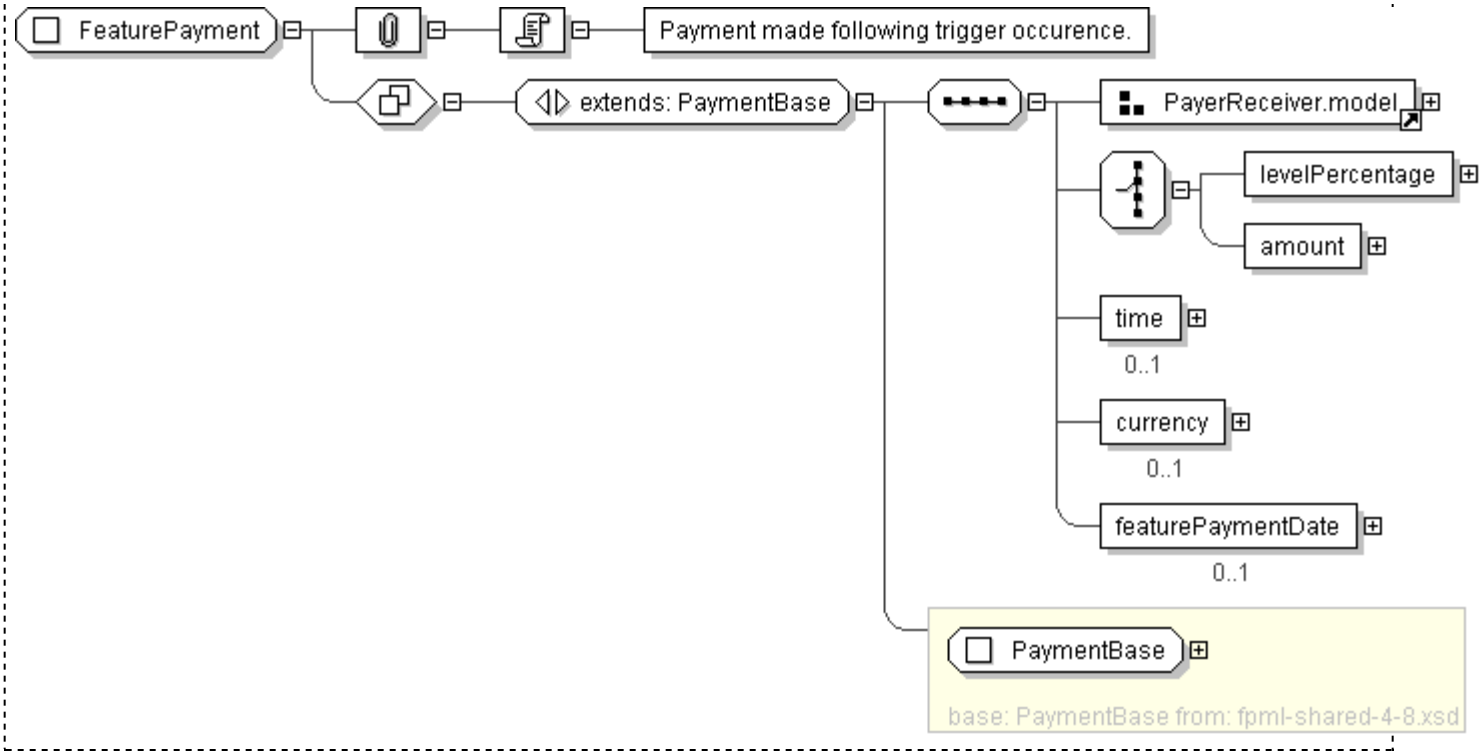
Super-types:	PaymentBase < FeaturePayment (by extension)
Sub-types:	None

Name	FeaturePayment
Used by (from the same schema document)	Complex Type TriggerEvent
Abstract	no
Documentation	Payment made following trigger occurence.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   Start Choice [1]
      <levelPercentage> xsd:decimal </levelPercentage> [1]
      'The trigger level percentage.'

      <amount> NonNegativeDecimal </amount> [1]
      'The monetary quantity in currency units.'

   End Choice
   <time> TimeTypeEnum </time> [0..1]
   'The feature payment time.'

   <currency> Currency </currency> [0..1]
   'The currency in which an amount is denominated.'

   <featurePaymentDate> AdjustableOrRelativeDate </featurePaymentDate> [0..1]
   'The feature payment date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FeaturePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:choice>
               <xsd:element name="levelPercentage" type=" xsd:decimal "/>
               <xsd:element name="amount" type=" NonNegativeDecimal "/>
            </xsd:choice>
            <xsd:element name="time" type=" TimeTypeEnum " minOccurs="0"/>
            <xsd:element name="currency" type=" Currency " minOccurs="0"/>
            <xsd:element name="featurePaymentDate" type=" AdjustableOrRelativeDate "
            minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FrequencyType

[Table of contents]

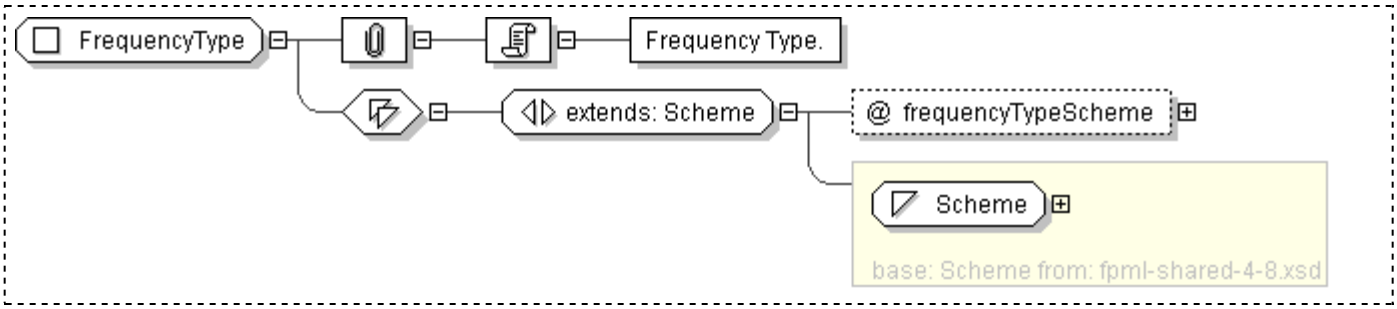
Super-types:	Scheme < FrequencyType (by extension)
Sub-types:	None

Name	FrequencyType
Used by (from the same schema document)	Complex Type AveragingSchedule
Abstract	no
Documentation	Frequency Type.

XML Instance Representation

```
<...
 frequencyTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FrequencyType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="frequencyTypeScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxFeature

[Table of contents]

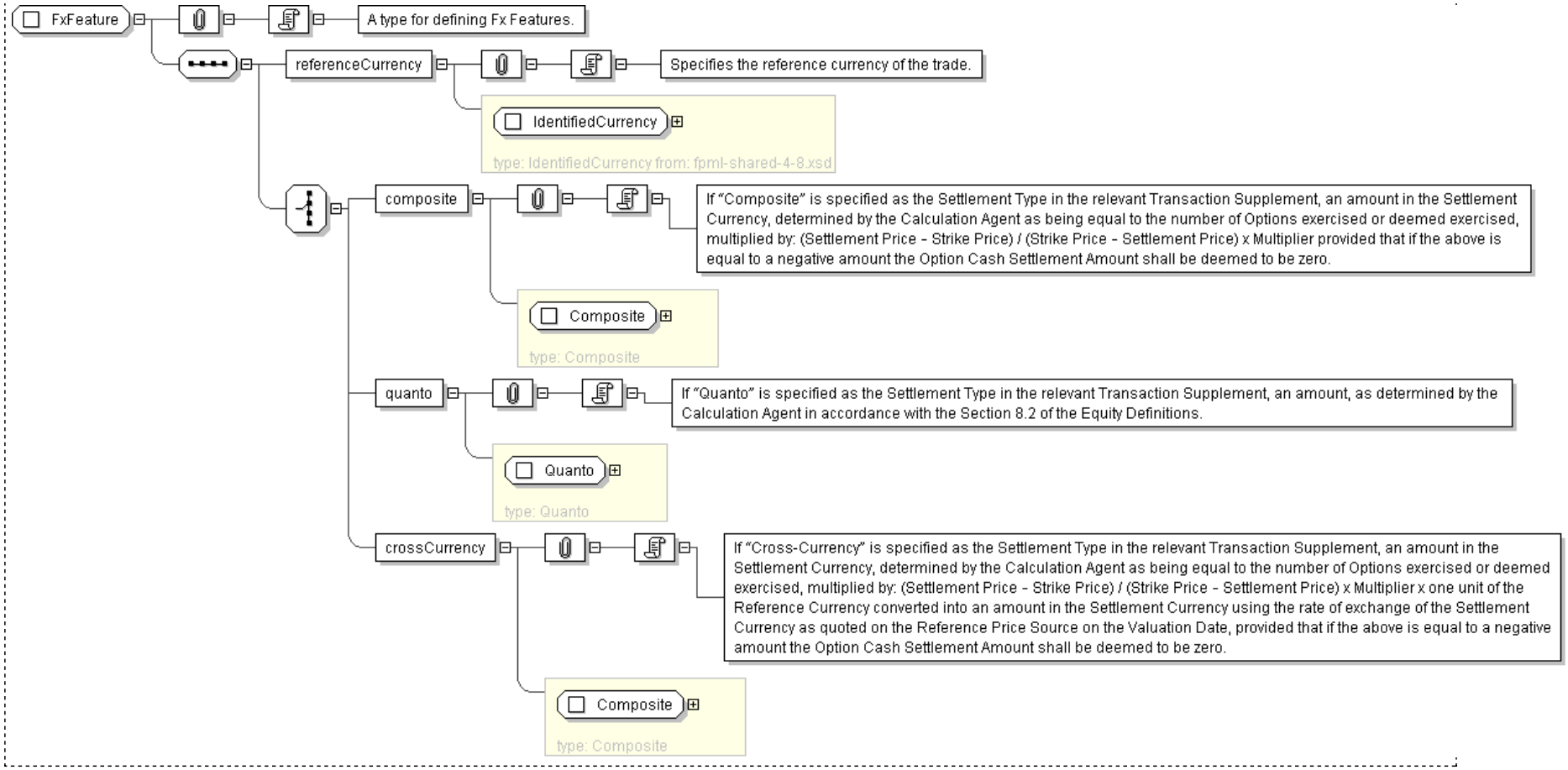
Super-types:	None
Sub-types:	None

Name	FxFeature
Used by (from the same schema document)	Model Group OptionBaseFeature.model
Abstract	no
Documentation	A type for defining Fx Features.

XML Instance Representation

```
<...>
  <referenceCurrency> IdentifiedCurrency </referenceCurrency> [1]
   'Specifies the reference currency of the trade.'

  Start Choice [1]
    <composite> Composite </composite> [1]
     'If â€œCompositeâ€ is specified as the Settlement Type in the relevant Transaction Supplement, an amount in the Settlement Currency,
     determined by the Calculation Agent as being equal to the number of Options exercised or deemed exercised, multiplied by: (Settlement
     Price â€“ Strike Price) / (Strike Price â€“ Settlement Price) x Multiplier provided that if the above is equal to a negative amount the
     Option Cash Settlement Amount shall be deemed to be zero.'

    <quanto> Quanto </quanto> [1]
     'If â€œQuantoâ€ is specified as the Settlement Type in the relevant Transaction Supplement, an amount, as determined by the Calculation
     Agent in accordance with the Section 8.2 of the Equity Definitions.'

    <crossCurrency> Composite </crossCurrency> [1]
     'If â€œCross-Currencyâ€ is specified as the Settlement Type in the relevant Transaction Supplement, an amount in the Settlement
     Currency, determined by the Calculation Agent as being equal to the number of Options exercised or deemed exercised, multiplied by:
     (Settlement Price â€“ Strike Price) / (Strike Price â€“ Settlement Price) x Multiplier x one unit of the Reference Currency converted
     into an amount in the Settlement Currency using the rate of exchange of the Settlement Currency as quoted on the Reference Price Source
     on the Valuation Date, provided that if the above is equal to a negative amount the Option Cash Settlement Amount shall be deemed to be
     zero.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxFeature">
  <xsd:sequence>
    <xsd:element name="referenceCurrency" type=" IdentifiedCurrency "/>
    <xsd:choice>
      <xsd:element name="composite" type=" Composite "/>
      <xsd:element name="quanto" type=" Quanto "/>
      <xsd:element name="crossCurrency" type=" Composite "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GracePeriodExtension

[Table of contents]

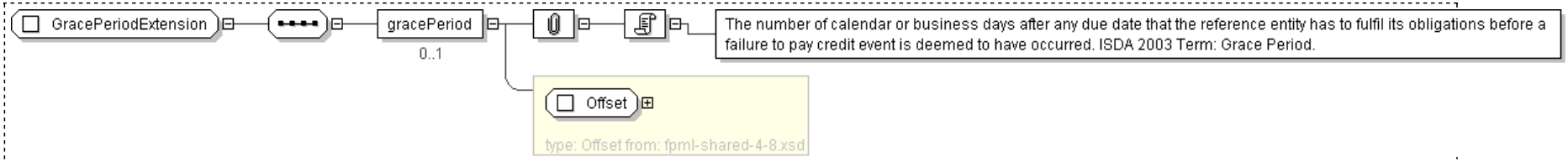
Super-types:	None
Sub-types:	None

Name	GracePeriodExtension
Used by (from the same schema document)	Complex Type FailureToPay
Abstract	no

XML Instance Representation

```
<...>
  <gracePeriod> Offset </gracePeriod> [0..1]
  'The number of calendar or business days after any due date that the reference entity has to fulfil its obligations before a failure to
  pay credit event is deemed to have occurred. ISDA 2003 Term: Grace Period.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GracePeriodExtension">
  <xsd:sequence>
    <xsd:element name="gracePeriod" type=" Offset " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Knock

[Table of contents]

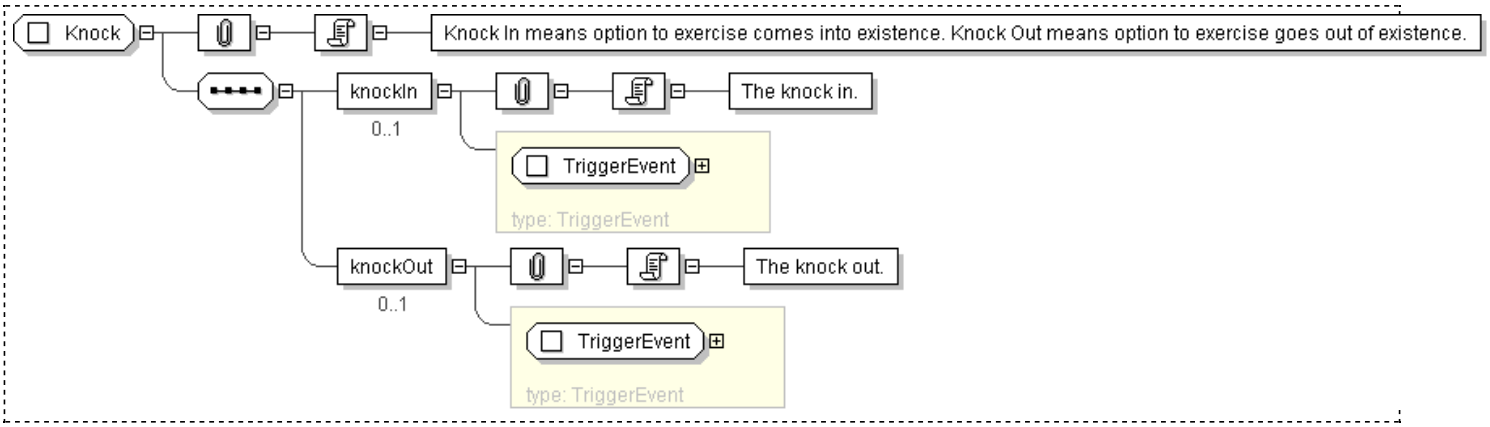
Super-types:	None
Sub-types:	None

Name	Knock
Used by (from the same schema document)	Model Group OptionFeature.model
Abstract	no
Documentation	Knock In means option to exercise comes into existence. Knock Out means option to exercise goes out of existence.

XML Instance Representation

```
<...>
   <knockIn> TriggerEvent </knockIn> [0..1]
    'The knock in.'

   <knockOut> TriggerEvent </knockOut> [0..1]
    'The knock out.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Knock">
   <xsd:sequence>
      <xsd:element name="knockIn" type=" TriggerEvent " minOccurs="0"/>
      <xsd:element name="knockOut" type=" TriggerEvent " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MarketDisruption

[Table of contents]

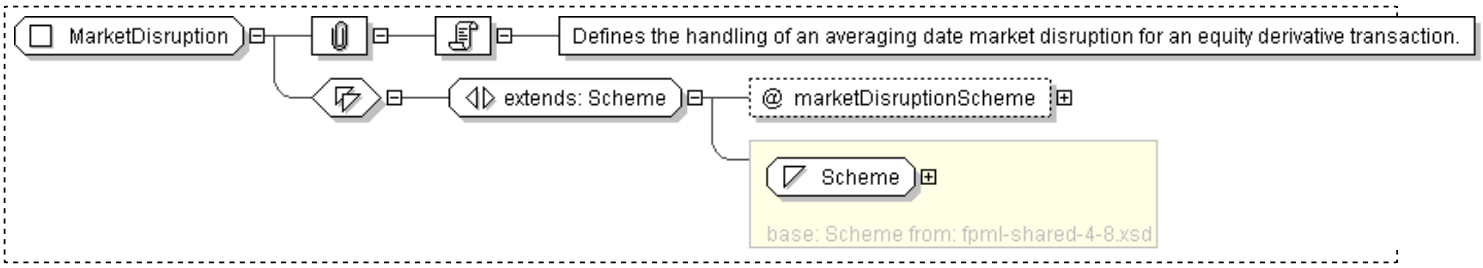
Super-types:	Scheme < MarketDisruption (by extension)
Sub-types:	None

Name	MarketDisruption
Used by (from the same schema document)	Complex Type AveragingPeriod
Abstract	no
Documentation	Defines the handling of an averaging date market disruption for an equity derivative transaction.

XML Instance Representation

```
<...
 marketDisruptionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MarketDisruption">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="marketDisruptionScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/market-disruption"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotifyingParty

[Table of contents]

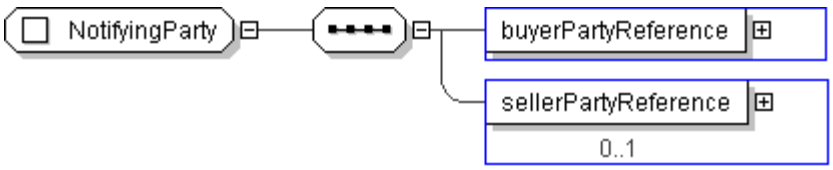
Super-types:	None
Sub-types:	None

Name	NotifyingParty
Used by (from the same schema document)	Complex Type CreditEventNotice
Abstract	no

XML Instance Representation

```
<...>
   <buyerPartyReference> PartyReference </buyerPartyReference> [1]
   <sellerPartyReference> PartyReference </sellerPartyReference> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotifyingParty">
   <xsd:sequence>
      <xsd:element name="buyerPartyReference" type=" PartyReference "/>
      <xsd:element name="sellerPartyReference" type=" PartyReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionBase

[Table of contents]

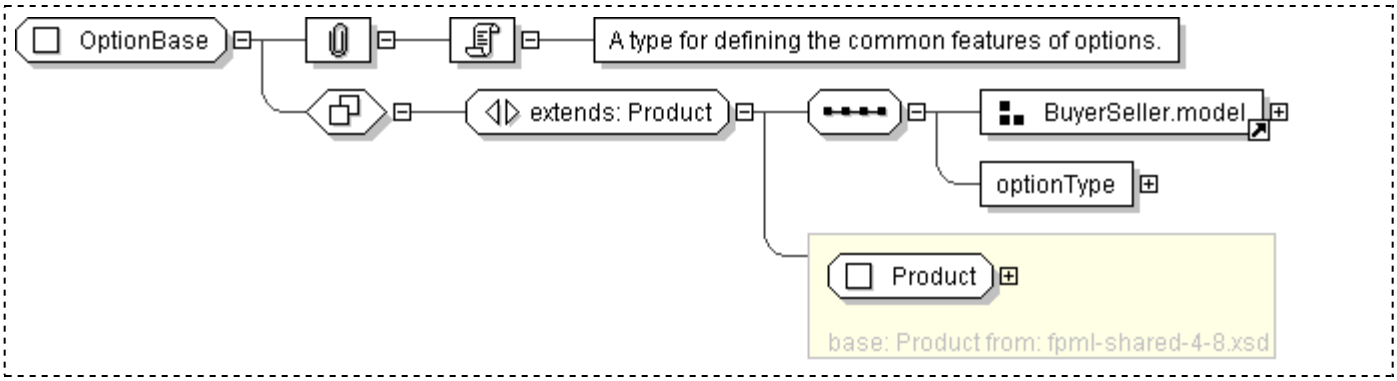
Super-types:	Product < OptionBase (by extension)
Sub-types:	• OptionBaseExtended (by extension)

Name	OptionBase
Abstract	yes
Documentation	A type for defining the common features of options.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this
    instrument and receives the rights defined by it. See 2000 ISDA definitions
    Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that
    grants the rights defined by this instrument and in return receives a payment
    for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is
    the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
    'The type of option transaction. From a usage standpoint, put/call is the
    default option type, while payer/receiver indicator is used for options index
    credit default swaps, consistently with the industry practice. Straddle is used
    for the case of straddle strategy, that combine a call and a put with the same
    strike.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionBase" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:group ref=" BuyerSeller.model "/>
        <xsd:element name="optionType" type=" OptionTypeEnum "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionBaseExtended

[Table of contents]

Super-types:	Product < OptionBase (by extension) < OptionBaseExtended (by extension)
Sub-types:	None

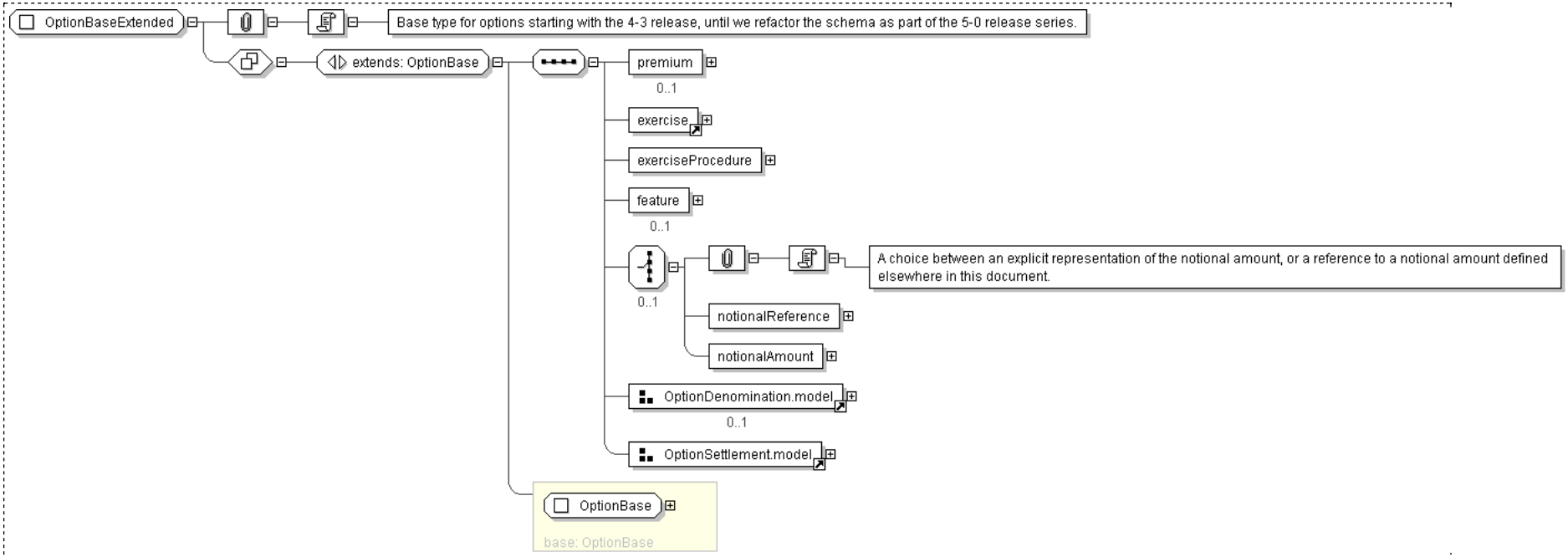
Name	OptionBaseExtended
Abstract	yes
Documentation	Base type for options starting with the 4-3 release, until we refactor the schema as part of the 5-0 release series.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
    'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA definitions Article
    11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
    'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment
    for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
    'The type of option transaction. From a usage standpoint, put/call is the default option type, while payer/receiver indicator is used for options index
    credit default swaps, consistently with the industry practice. Straddle is used for the case of straddle strategy, that combine a call and a put with the
    same strike.'

   <premium> Premium </premium> [0..1]
    'The option premium payable by the buyer to the seller.'

   <exercise> ... </exercise> [1]
   <exerciseProcedure> ExerciseProcedure </exerciseProcedure> [1]
    'A set of parameters defining procedures associated with the exercise.'

   <feature> OptionFeature </feature> [0..1]
    'An Option feature such as quanto, asian, barrier, knock.'

   Start Choice [0..1]
'A choice between an explicit representation of the notional amount, or a reference to a notional amount defined elsewhere in this document.'


      <notionalReference> NotionalAmountReference </notionalReference> [1]
      <notionalAmount> Money </notionalAmount> [1]
   End Choice
   Start Group: OptionDenomination.model [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
       'The number of units of underlyer per option comprised in the option transaction.'

      <entitlementCurrency> Currency </entitlementCurrency> [0..1]
       'TODO'

      <numberOfOptions> PositiveDecimal </numberOfOptions> [0..1]
       'The number of options comprised in the option transaction.'

   End Group: OptionDenomination.model
   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
          'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
          'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
```

```
  End Group: SettlementAmountOrCurrency.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionBaseExtended" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" OptionBase ">
         <xsd:sequence>
            <xsd:element name="premium" type=" Premium " minOccurs="0"/>
            <xsd:element ref=" exercise "/>
            <xsd:element name="exerciseProcedure" type=" ExerciseProcedure "/>
            <xsd:element name="feature" type=" OptionFeature " minOccurs="0"/>
            <xsd:choice minOccurs="0">
               <xsd:element name="notionalReference" type=" NotionalAmountReference "/>
               <xsd:element name="notionalAmount" type=" Money "/>
            </xsd:choice>
            <xsd:group ref=" OptionDenomination.model " minOccurs="0"/>
            <xsd:group ref=" OptionSettlement.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionFeature

[Table of contents]

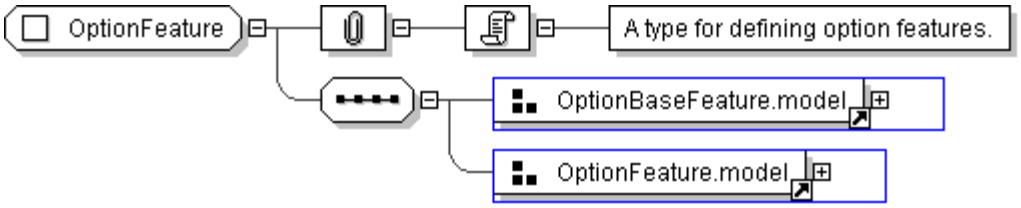
Super-types:	None
Sub-types:	None

Name	OptionFeature
Used by (from the same schema document)	Complex Type OptionBaseExtended
Abstract	no
Documentation	A type for defining option features.

XML Instance Representation

```
<...>
   <fxFeature> FxFeature </fxFeature> [0..1]
    'A quanto or composite FX feature.'

   <strategyFeature> StrategyFeature </strategyFeature> [0..1]
    'A simple strategy feature.'

   <asian> Asian </asian> [0..1]
    'An option where and average price is taken on valuation.'

   <barrier> Barrier </barrier> [0..1]
    'An option with a barrier feature.'

   <knock> Knock </knock> [0..1]
    'A knock feature.'

   <passThrough> PassThrough </passThrough> [0..1]
    'Pass through payments from the underlyer, such as dividends.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionFeature">
   <xsd:sequence>
      <xsd:group ref=" OptionBaseFeature.model "/>
      <xsd:group ref=" OptionFeature.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionNumericStrike

[Table of contents]

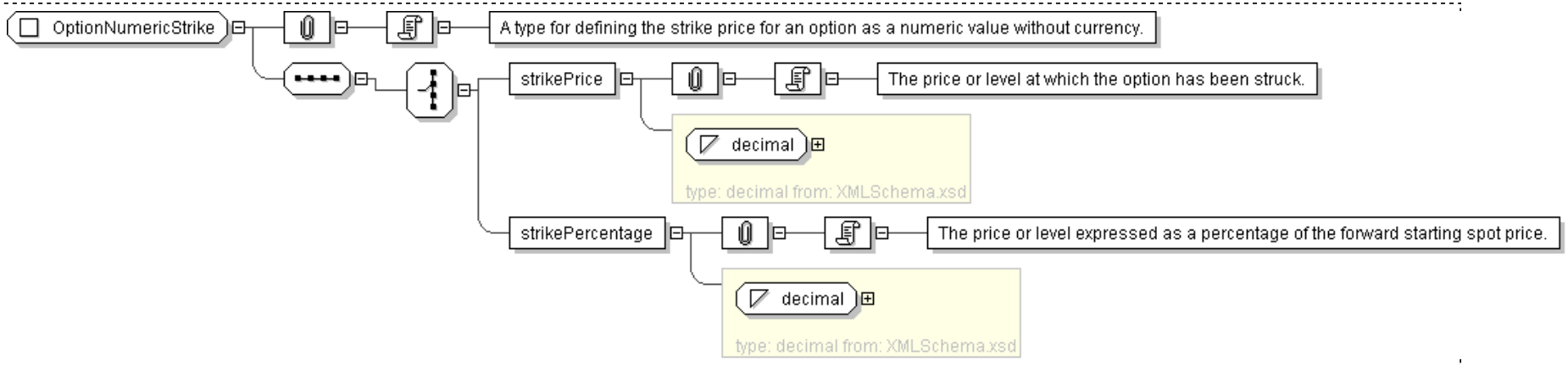
Super-types:	None
Sub-types:	• OptionStrike (by extension)

Name	OptionNumericStrike
Abstract	no
Documentation	A type for defining the strike price for an option as a numeric value without currency.

XML Instance Representation

```
<...>
  Start Choice [1]
    <strikePrice> xsd:decimal </strikePrice> [1]
    'The price or level at which the option has been struck.'

    <strikePercentage> xsd:decimal </strikePercentage> [1]
    'The price or level expressed as a percentage of the forward starting spot price.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionNumericStrike">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="strikePrice" type=" xsd:decimal "/>
      <xsd:element name="strikePercentage" type=" xsd:decimal "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OptionStrike

[Table of contents]

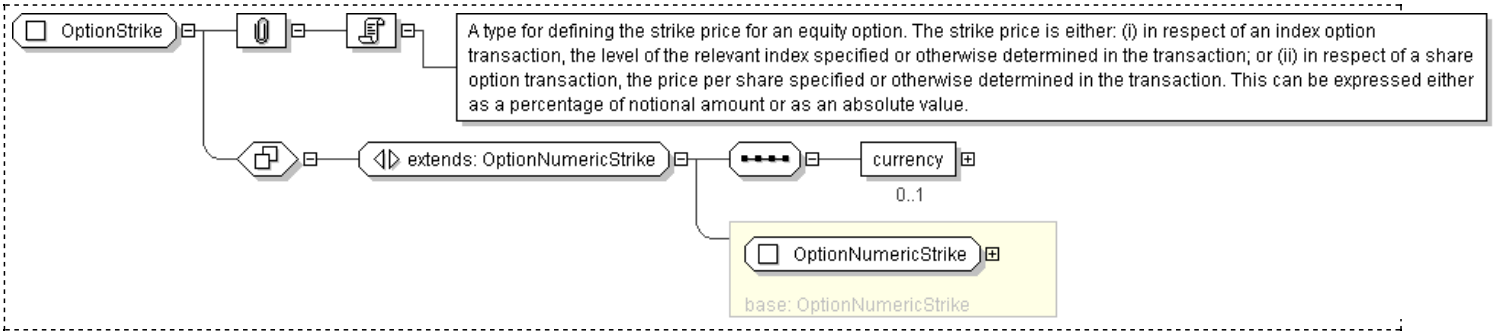
Super-types:	OptionNumericStrike < OptionStrike (by extension)
Sub-types:	None

Name	OptionStrike
Used by (from the same schema document)	Complex Type StrikeSpread
Abstract	no
Documentation	A type for defining the strike price for an equity option. The strike price is either: (i) in respect of an index option transaction, the level of the relevant index specified or otherwise determined in the transaction; or (ii) in respect of a share option transaction, the price per share specified or otherwise determined in the transaction. This can be expressed either as a percentage of notional amount or as an absolute value.

XML Instance Representation

```
<...>
  Start Choice [1]
    <strikePrice> xsd:decimal </strikePrice> [1]
     'The price or level at which the option has been struck.'

    <strikePercentage> xsd:decimal </strikePercentage> [1]
     'The price or level expressed as a percentage of the forward starting spot price.'

  End Choice
  <currency> Currency </currency> [0..1]
   'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionStrike">
  <xsd:complexContent>
    <xsd:extension base=" OptionNumericStrike ">
      <xsd:sequence>
        <xsd:element name="currency" type=" Currency " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PassThrough

[Table of contents]

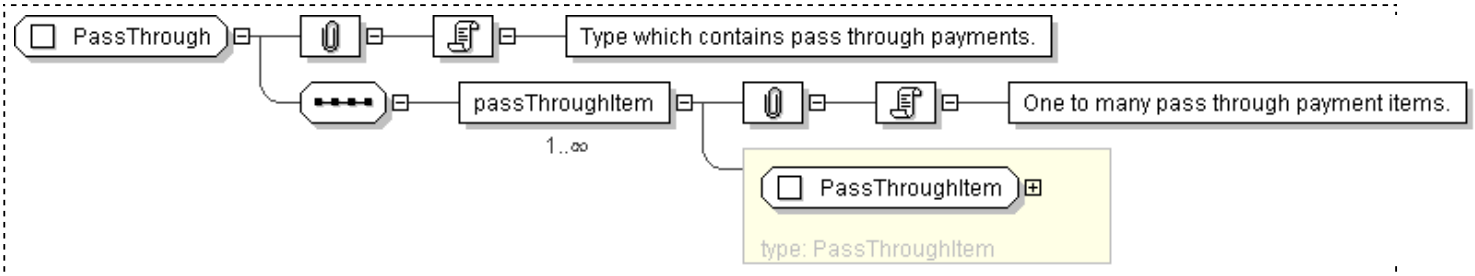
Super-types:	None
Sub-types:	None

Name	PassThrough
Used by (from the same schema document)	Model Group OptionFeature.model
Abstract	no
Documentation	Type which contains pass through payments.

XML Instance Representation

```
<...>
   <passThroughItem> PassThroughItem </passThroughItem> [1..*]
   'One to many pass through payment items.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PassThrough">
   <xsd:sequence>
      <xsd:element name="passThroughItem" type=" PassThroughItem " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PassThroughItem

[Table of contents]

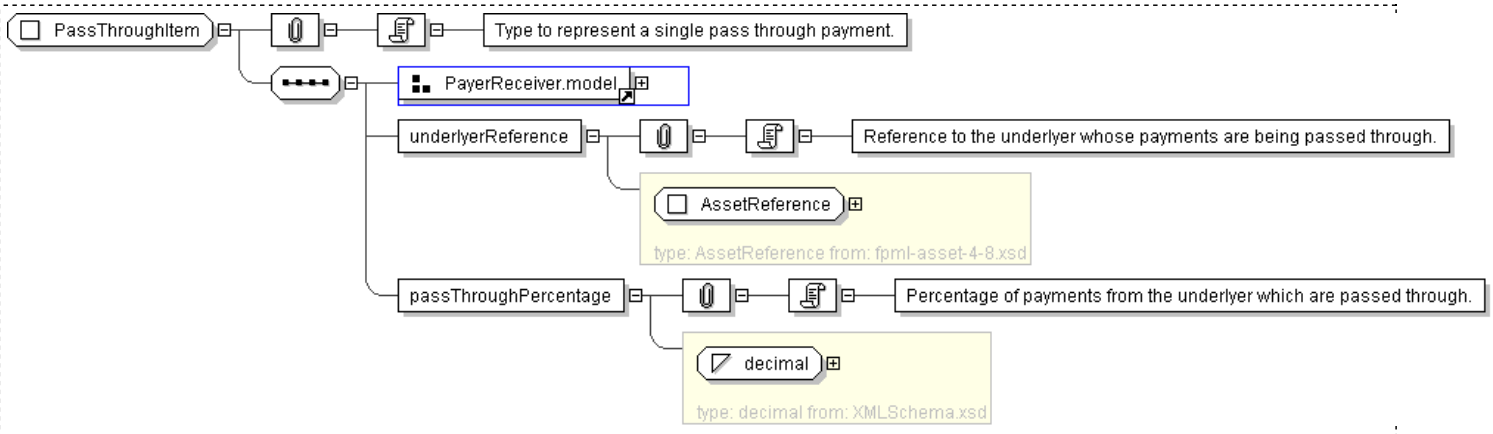
Super-types:	None
Sub-types:	None

Name	PassThroughItem
Used by (from the same schema document)	Complex Type PassThrough
Abstract	no
Documentation	Type to represent a single pass through payment.

XML Instance Representation

```
<...>
  <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
  'A reference to the party responsible for making the payments defined by this structure.'

  <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
  'A reference to the party that receives the payments corresponding to this structure.'

  <underlyerReference> AssetReference </underlyerReference> [1]
  'Reference to the underlyer whose payments are being passed through.'

  <passThroughPercentage> xsd:decimal </passThroughPercentage> [1]
  'Percentage of payments from the underlyer which are passed through.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PassThroughItem">
  <xsd:sequence>
    <xsd:group ref=" PayerReceiver.model "/>
    <xsd:element name="underlyerReference" type=" AssetReference "/>
    <xsd:element name="passThroughPercentage" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Premium

[Table of contents]

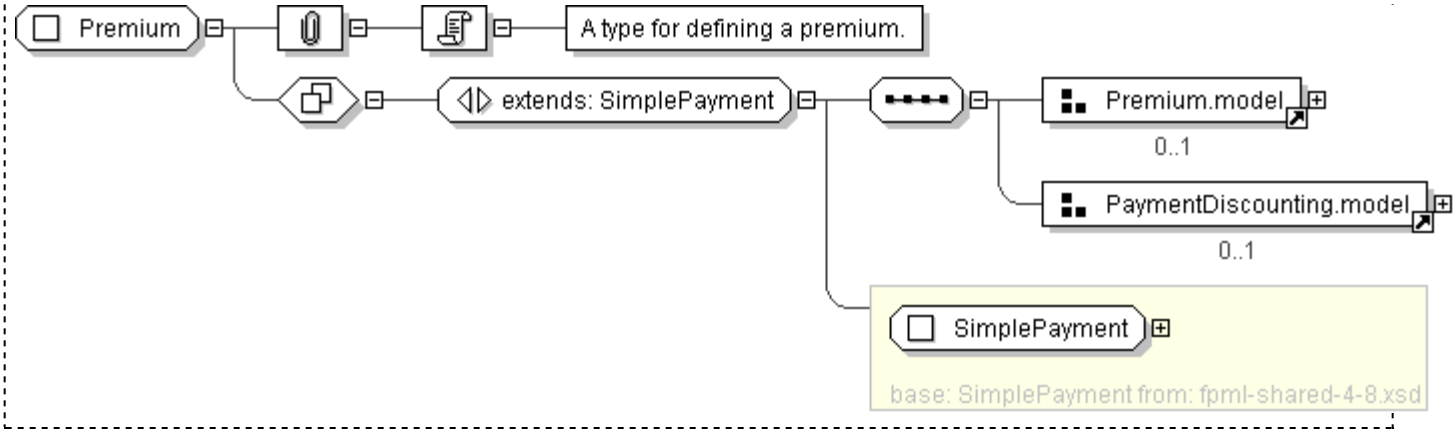
Super-types:	SimplePayment < Premium (by extension)
Sub-types:	None

Name	Premium
Used by (from the same schema document)	Complex Type OptionBaseExtended
Abstract	no
Documentation	A type for defining a premium.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this
    structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this
    structure.'

   <paymentAmount> Money </paymentAmount> [1]
   <paymentDate> AdjustableOrRelativeAndAdjustedDate </paymentDate> [1]
    'The payment date. This date is subject to adjustment in accordance with any
    applicable business day convention.'

   Start Group: Premium.model [0..1]
      <premiumType> PremiumTypeEnum </premiumType> [0..1]
       'Forward start Premium type'

      <pricePerOption> Money </pricePerOption> [0..1]
       'The amount of premium to be paid expressed as a function of the number of
       options.'

      <percentageOfNotional> xsd:decimal </percentageOfNotional> [0..1]
       'The amount of premium to be paid expressed as a percentage of the notional
       value of the transaction. A percentage of 5% would be expressed as 0.05.'

   End Group: Premium.model
   Start Group: PaymentDiscounting.model [0..1]
      <discountFactor> xsd:decimal </discountFactor> [0..1]
       'The value representing the discount factor used to calculate the present
       value of the cash flow.'

      <presentValueAmount> Money </presentValueAmount> [0..1]
       'The amount representing the present value of the forecast payment.'

   End Group: PaymentDiscounting.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Premium">
   <xsd:complexContent>
      <xsd:extension base=" SimplePayment ">
         <xsd:sequence>
            <xsd:group ref=" Premium.model " minOccurs="0"/>
            <xsd:group ref=" PaymentDiscounting.model " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PubliclyAvailableInformation

[Table of contents]

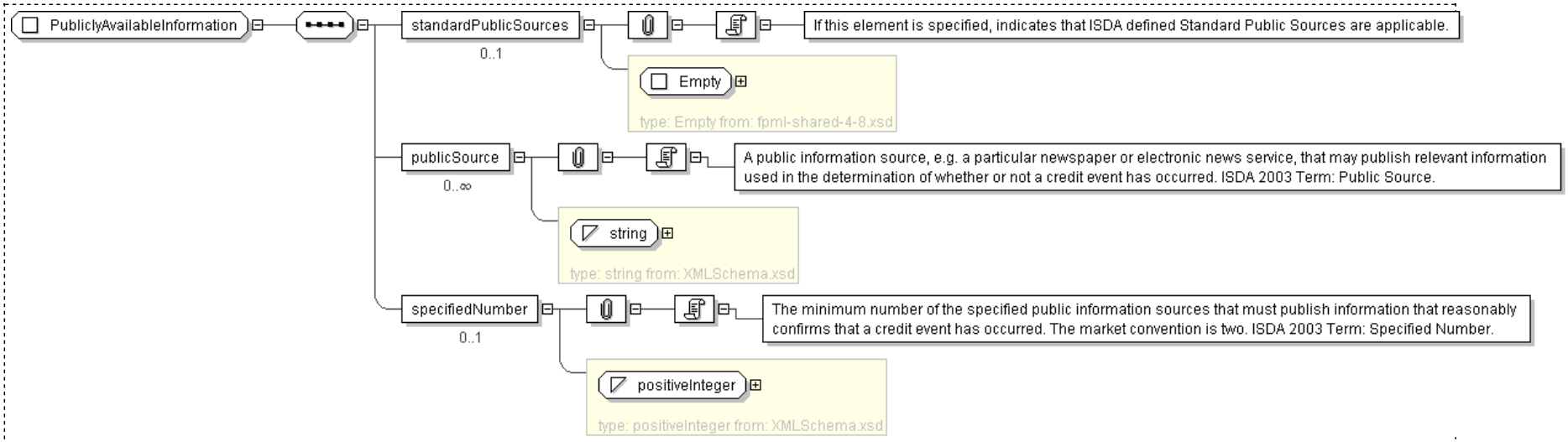
Super-types:	None
Sub-types:	None

Name	PubliclyAvailableInformation
Used by (from the same schema document)	Complex Type CreditEventNotice
Abstract	no

XML Instance Representation

```
<...>
   <standardPublicSources> Empty </standardPublicSources> [0..1]
   'If this element is specified, indicates that ISDA defined Standard Public Sources are applicable.'

   <publicSource> xsd:string </publicSource> [0..*]
   'A public information source, e.g. a particular newspaper or electronic news service, that may publish relevant information used in the
   determination of whether or not a credit event has occurred. ISDA 2003 Term: Public Source.'

   <specifiedNumber> xsd:positiveInteger </specifiedNumber> [0..1]
   'The minimum number of the specified public information sources that must publish information that reasonably confirms that a credit event has
   occurred. The market convention is two. ISDA 2003 Term: Specified Number.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PubliclyAvailableInformation">
   <xsd:sequence>
      <xsd:element name="standardPublicSources" type=" Empty " minOccurs="0"/>
      <xsd:element name="publicSource" type=" xsd:string " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="specifiedNumber" type=" xsd:positiveInteger " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Quanto

[Table of contents]

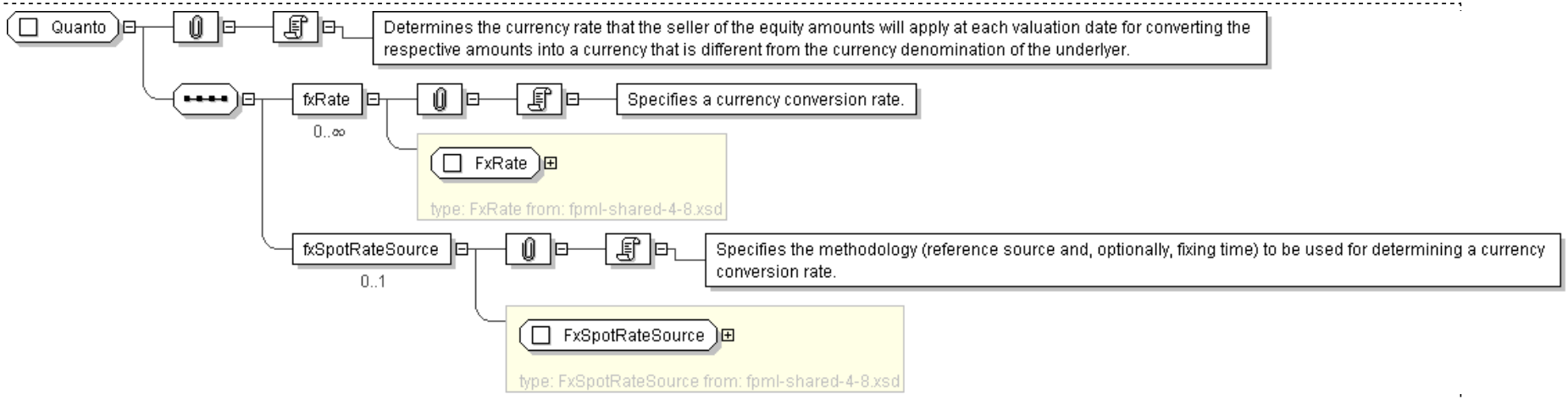
Super-types:	None
Sub-types:	None

Name	Quanto
Used by (from the same schema document)	Complex Type ExFeature
Abstract	no
Documentation	Determines the currency rate that the seller of the equity amounts will apply at each valuation date for converting the respective amounts into a currency that is different from the currency denomination of the underlyer.

XML Instance Representation

```
<...>
   <fxRate> FxRate </fxRate> [0..*]
   'Specifies a currency conversion rate.'

   <fxSpotRateSource> FxSpotRateSource </fxSpotRateSource> [0..1]
   'Specifies the methodology (reference source and, optionally, fixing time) to be used for determining a currency conversion rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Quanto">
   <xsd:sequence>
      <xsd:element name="fxRate" type=" FxRate " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="fxSpotRateSource" type=" FxSpotRateSource " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Restructuring

[Table of contents]

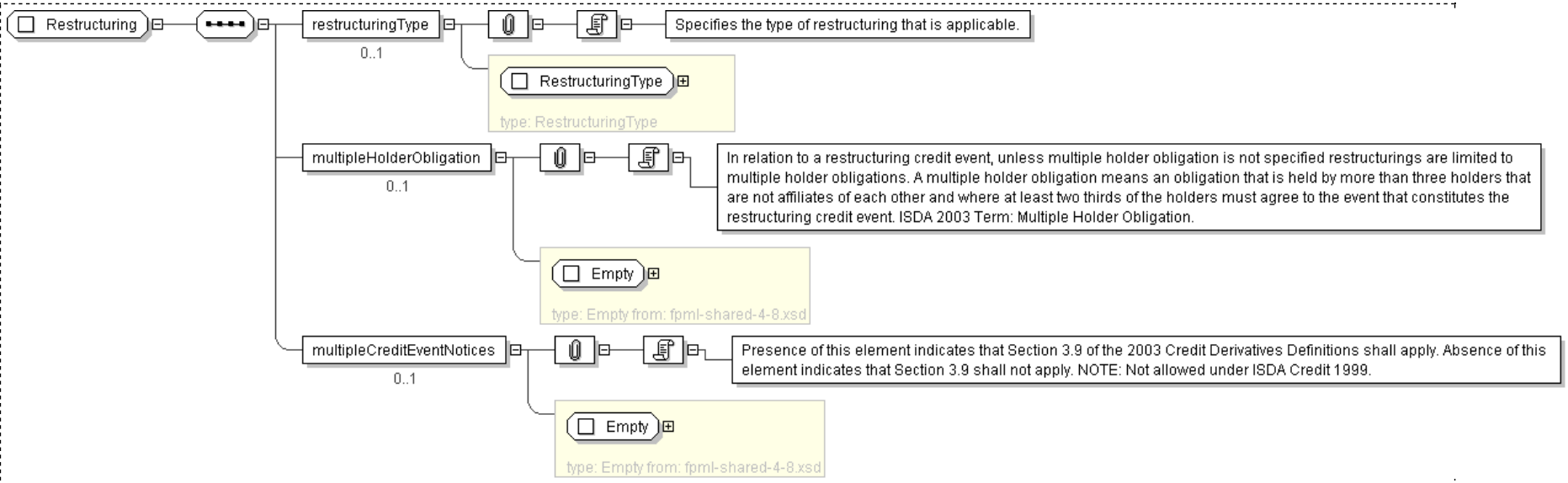
Super-types:	None
Sub-types:	None

Name	Restructuring
Used by (from the same schema document)	Complex Type CreditEvents
Abstract	no

XML Instance Representation

```
<...>
  <restructuringType> RestructuringType </restructuringType> [0..1]
  'Specifies the type of restructuring that is applicable.'

  <multipleHolderObligation> Empty </multipleHolderObligation> [0..1]
  'In relation to a restructuring credit event, unless multiple holder obligation is not specified restructurings are limited to multiple
  holder obligations. A multiple holder obligation means an obligation that is held by more than three holders that are not affiliates of each
  other and where at least two thirds of the holders must agree to the event that constitutes the restructuring credit event. ISDA 2003 Term:
  Multiple Holder Obligation.'

  <multipleCreditEventNotices> Empty </multipleCreditEventNotices> [0..1]
  'Presence of this element indicates that Section 3.9 of the 2003 Credit Derivatives Definitions shall apply. Absence of this element
  indicates that Section 3.9 shall not apply. NOTE: Not allowed under ISDA Credit 1999.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Restructuring">
  <xsd:sequence>
    <xsd:element name="restructuringType" type=" RestructuringType " minOccurs="0"/>
    <xsd:element name="multipleHolderObligation" type=" Empty " minOccurs="0"/>
```

```
    <xsd:element name="multipleCreditEventNotices" type=" Empty " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RestructuringType

[Table of contents]

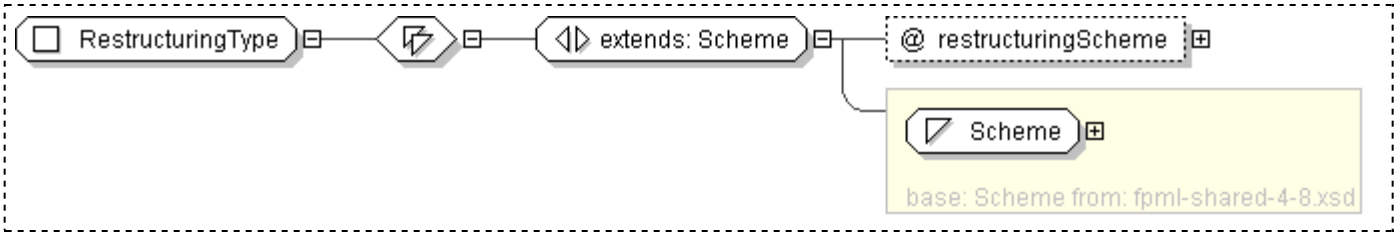
Super-types:	Scheme < RestructuringType (by extension)
Sub-types:	None

Name	RestructuringType
Used by (from the same schema document)	Complex Type Restructuring
Abstract	no

XML Instance Representation

```
<...
 restructuringScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RestructuringType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="restructuringScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/restructuring"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StrategyFeature

[Table of contents]

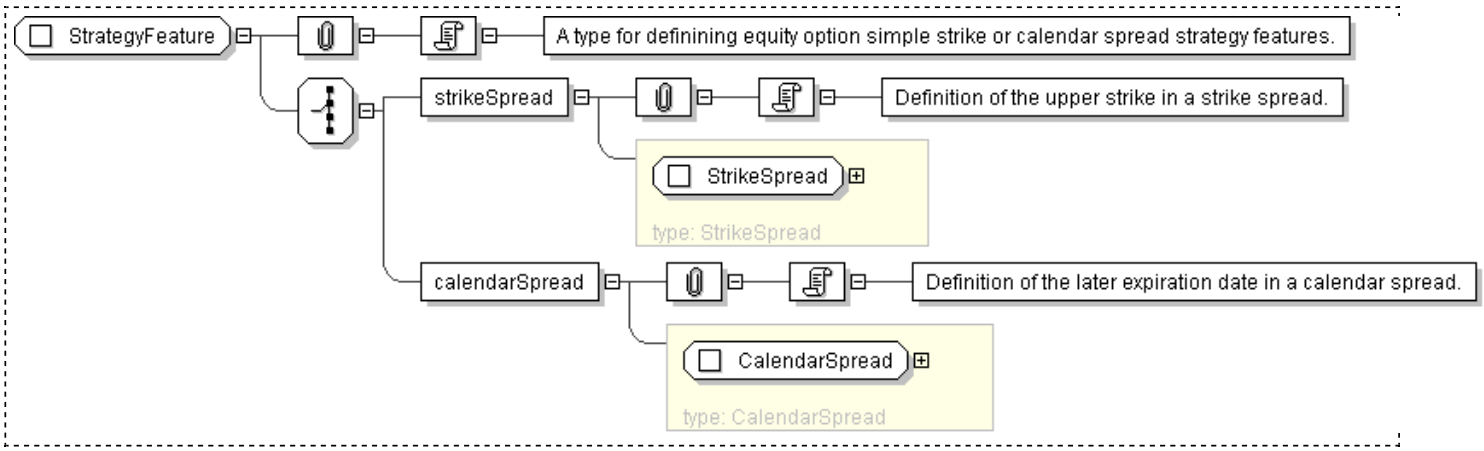
Super-types:	None
Sub-types:	None

Name	StrategyFeature
Used by (from the same schema document)	Model Group OptionBaseFeature.model
Abstract	no
Documentation	A type for definining equity option simple strike or calendar spread strategy features.

XML Instance Representation

```
<...>
  Start Choice [1]
    <strikeSpread> StrikeSpread </strikeSpread> [1]
    'Definition of the upper strike in a strike spread.'

    <calendarSpread> CalendarSpread </calendarSpread> [1]
    'Definition of the later expiration date in a calendar spread.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StrategyFeature">
  <xsd:choice>
    <xsd:element name="strikeSpread" type=" StrikeSpread "/>
    <xsd:element name="calendarSpread" type=" CalendarSpread "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StrikeSpread

[Table of contents]

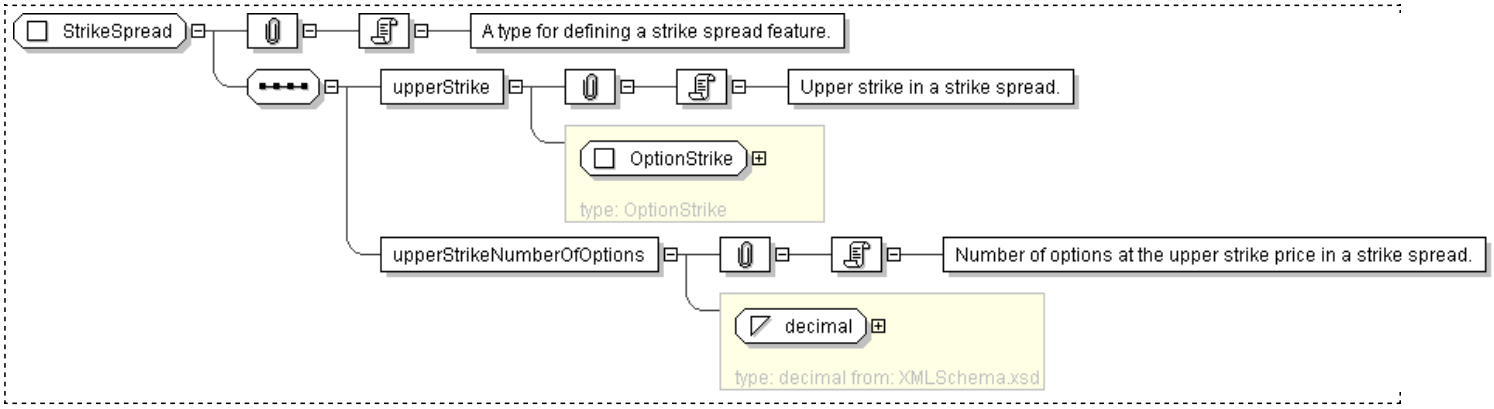
Super-types:	None
Sub-types:	None

Name	StrikeSpread
Used by (from the same schema document)	Complex Type StrategyFeature
Abstract	no
Documentation	A type for defining a strike spread feature.

XML Instance Representation

```
<...>
  <upperStrike> OptionStrike </upperStrike> [1]
  'Upper strike in a strike spread.'

  <upperStrikeNumberOfOptions> xsd:decimal </upperStrikeNumberOfOptions> [1]
  'Number of options at the upper strike price in a strike spread.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StrikeSpread">
  <xsd:sequence>
    <xsd:element name="upperStrike" type=" OptionStrike "/>
    <xsd:element name="upperStrikeNumberOfOptions" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Trigger

[Table of contents]

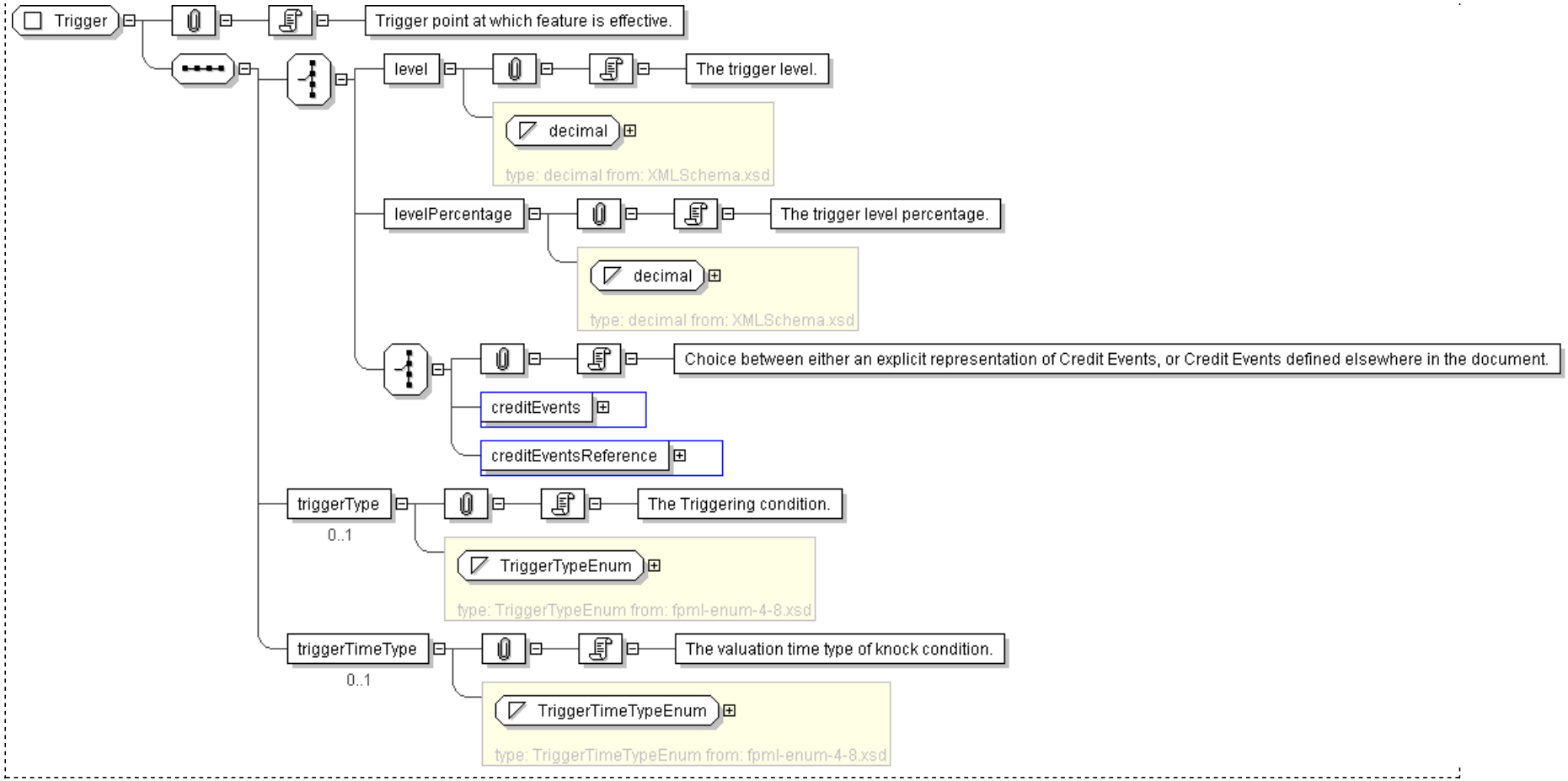
Super-types:	None
Sub-types:	None

Name	Trigger
Used by (from the same schema document)	Complex Type TriggerEvent
Abstract	no
Documentation	Trigger point at which feature is effective.

XML Instance Representation

```
<...>
   Start Choice [1]
      <level> xsd:decimal </level> [1]
      'The trigger level.'

      <levelPercentage> xsd:decimal </levelPercentage> [1]
      'The trigger level percentage.'

      Start Choice [1]
'Choice between either an explicit representation of Credit Events, or Credit Events defined elsewhere in the document.'


         <creditEvents> CreditEvents </creditEvents> [1]
         <creditEventsReference> CreditEventsReference </creditEventsReference> [1]
      End Choice
   End Choice
   <triggerType> TriggerTypeEnum </triggerType> [0..1]
   'The Triggering condition.'

   <triggerTimeType> TriggerTimeTypeEnum </triggerTimeType> [0..1]
   'The valuation time type of knock condition.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Trigger">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="level" type=" xsd:decimal "/>
      <xsd:element name="levelPercentage" type=" xsd:decimal "/>
      <xsd:choice>
        <xsd:element name="creditEvents" type=" CreditEvents "/>
        <xsd:element name="creditEventsReference" type=" CreditEventsReference "/>
      </xsd:choice>
    </xsd:choice>
    <xsd:element name="triggerType" type=" TriggerTypeEnum " minOccurs="0" maxOccurs="1"/>
    <xsd:element name="triggerTimeType" type=" TriggerTimeTypeEnum " minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TriggerEvent

[Table of contents]

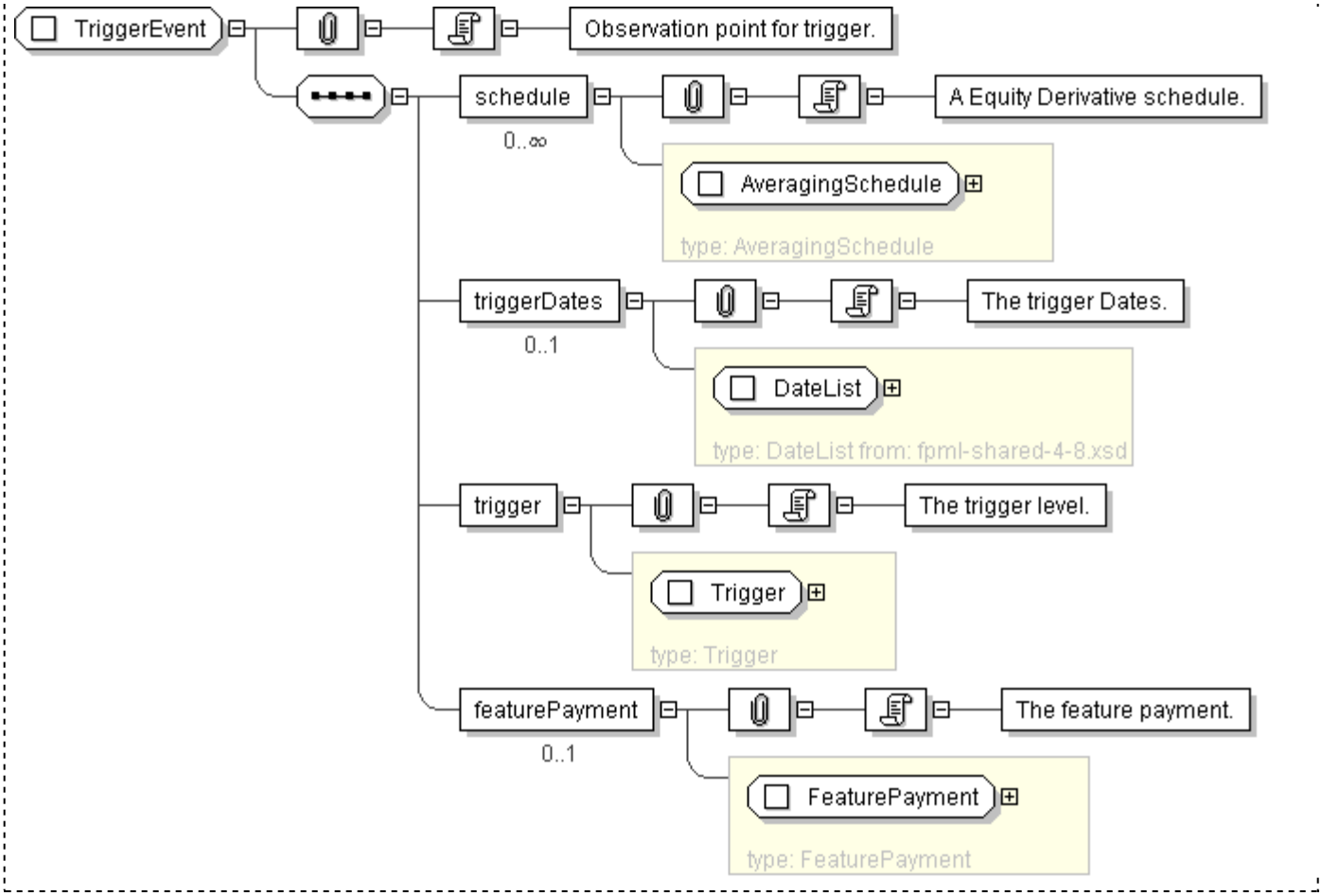
Super-types:	None
Sub-types:	None

Name	TriggerEvent
Used by (from the same schema document)	Complex Type Barrier , Complex Type Barrier , Complex Type Knock , Complex Type Knock
Abstract	no
Documentation	Observation point for trigger.

XML Instance Representation

```
<...>
   <schedule> AveragingSchedule </schedule> [0..*]
   'A Equity Derivative schedule.'

   <triggerDates> DateList </triggerDates> [0..1]
   'The trigger Dates.'

   <trigger> Trigger </trigger> [1]
   'The trigger level.'

   <featurePayment> FeaturePayment </featurePayment> [0..1]
   'The feature payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TriggerEvent">
  <xsd:sequence>
    <xsd:element name="schedule" type=" AveragingSchedule " minOccurs="0"
    maxOccurs="unbounded"/>
    <xsd:element name="triggerDates" type=" DateList " minOccurs="0"/>
    <xsd:element name="trigger" type=" Trigger "/>
    <xsd:element name="featurePayment" type=" FeaturePayment " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: WeightedAveragingObservation

[Table of contents]

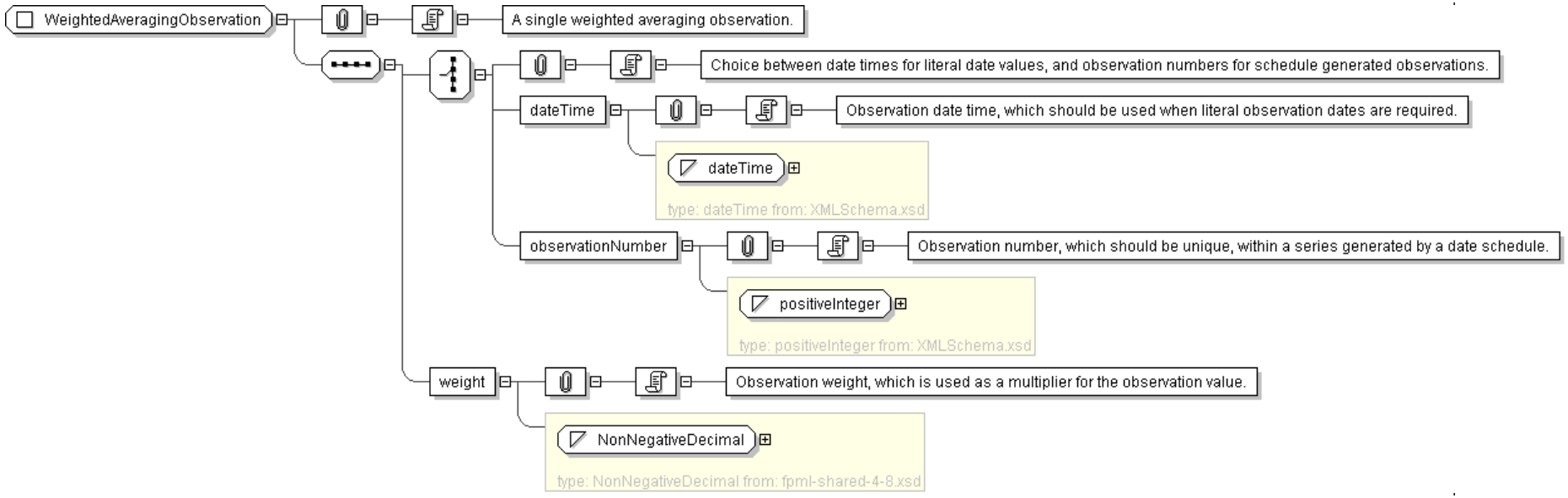
Super-types:	None
Sub-types:	None

Name	WeightedAveragingObservation
Used by (from the same schema document)	Complex Type AveragingObservationList
Abstract	no
Documentation	A single weighted averaging observation.

XML Instance Representation

```
<...>
  Start Choice [1]
'Choice between date times for literal date values, and observation numbers for schedule generated observations.'

    <dateTime> xsd:dateTime </dateTime> [1]
    'Observation date time, which should be used when literal observation dates are required.'

    <observationNumber> xsd:positiveInteger </observationNumber> [1]
    'Observation number, which should be unique, within a series generated by a date schedule.'

  End Choice
  <weight> NonNegativeDecimal </weight> [1]
  'Observation weight, which is used as a multiplier for the observation value.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="WeightedAveragingObservation">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="dateTime" type=" xsd:dateTime "/>
         <xsd:element name="observationNumber" type=" xsd:positiveInteger "/>
      </xsd:choice>
      <xsd:element name="weight" type=" NonNegativeDecimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: AffectedTransactions
  - Complex Type: Novation
  - Complex Type: NovationNotificationMessage
  - Complex Type: NovationRequestMessage
  - Complex Type: NovationResponseMessage
  - Complex Type: PartialTerminationAmount
  - Complex Type: Termination
  - Complex Type: TradeAmendment
  - Model Group: NovationDetails.model
  - Model Group: NovationMessage.model
  - Model Group: TerminationDetails.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml-annotation	http://www.fpml.org/annotation
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
```

```
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
```

```
                </restriction>
              </simpleType>
            </element>
          </sequence>
          <attribute name="country" type=" string " fixed="Australia"/>
        </extension>
    </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See:

http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: NovationDetails.model

[Table of contents]

Name	NovationDetails.model
Used by (from the same schema document)	Complex Type Novation

XML Instance Representation

```
Start Choice [1]
   Start Choice [1]
      <newTransactionReference> PartyTradeIdentifiers </newTransactionReference> [1]
      'Indicates a reference to the new transaction between the transferee and the remaining party.'

      <newTransaction> Trade </newTransaction> [1]
      'Indicates the new transaction between the transferee and the remaining party.'

   End Choice
   Start Choice [1]
      <oldTransactionReference> PartyTradeIdentifiers </oldTransactionReference> [1]
      'Indicates a reference to the original trade between the transferor and the remaining party.'

      <oldTransaction> Trade </oldTransaction> [1]
      'Indicates the original trade between the transferor and the remaining party.'

   End Choice
   Start Choice [0..1]
      <newTransactionReference> PartyTradeIdentifiers </newTransactionReference> [1]
      'Indicates a reference to the new transaction between the transferee and the remaining party.'

      <newTransaction> Trade </newTransaction> [1]
      'Indicates the new transaction between the transferee and the remaining party.'

   End Choice
End Choice
<transferor> PartyReference </transferor> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferor (outgoing party) in the novation. The Transferor means a party which transfers by novation to a Transferee all of its rights, liabilities, duties and obligations with respect to a Remaining Party. In a four-way novation the party referenced is Transferor 1 which transfers by novation to Transferee 1 all of its rights, liabilities, duties and obligations with respect to Transferor 2. ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-way novation).'

<transferee> PartyReference </transferee> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferee (incoming party) in the novation. Transferee means a party which accepts by way of novation all rights, liabilities, duties and obligations of a Transferor with respect to a Remaining Party. In a four-way novation the party referenced is Transferee 1 which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).'

<remainingParty> PartyReference </remainingParty> [1]
'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the novation. Remaining Party means a party which consents to a Transferor\'s transfer by novation and the acceptance thereof by the Transferee of all of the Transferor\'s rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation) or Transferor 2 (four-way novation).'

<otherRemainingParty> PartyReference </otherRemainingParty> [0..1]
'A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).'

<novationDate> xsd:date </novationDate> [1]
'Specifies the date that one party\'s legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of the 2004 ISDA Novation Definitions, section 1.16.'

<novationTradeDate> xsd:date </novationTradeDate> [0..1]
'Specifies the date the parties agree to assign or novate a trade. If this element is not specified, the novationTradeDate will be deemed to be the novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.'

Start Choice [0..1]
   <novatedAmount> Money </novatedAmount> [1]
   'The amount which represents the portion of the Old Transaction being novated.'

   <novatedNumberOfOptions> xsd:decimal </novatedNumberOfOptions> [1]
   'The number of options which represent the portion of the Old Transaction being novated.'

End Choice
<remainingTrade> Trade </remainingTrade> [0..1]
'This element contains a description of the remaining portion of a partially novated trade.'
```

<fullFirstCalculationPeriod> xsd:boolean </fullFirstCalculationPeriod> [0..1]
'This element corresponds to the applicability of the Full First Calculation Period as defined in the 2004 ISDA Novation Definitions, section 1.20.'

<firstPeriodStartDate> FirstPeriodStartDate </firstPeriodStartDate> [0..2]
'Element that is used to be able to make sense of the â€œnew transactionâ€ without requiring reference back to the â€œold transactionâ€. In the case of interest rate products there are potentially 2 â€œfirst period start datesâ€ to reference â€“ one with respect to each party to the new transaction. For Credit Default Swaps there is just the one with respect to the party that is the fixed rate payer.'

<nonReliance> Empty </nonReliance> [0..1]
'This element corresponds to the non-Reliance section in the 2004 ISDA Novation Definitions, section 2.1 (c) (i). The element appears in the instance document when non-Reliance is applicable.'

<creditDerivativesNotices> CreditDerivativesNotices </creditDerivativesNotices> [0..1]
'This element should be specified if one or more of either a Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party. The type of notice or notices that have been delivered should be indicated by setting the relevant boolean element value(s) to true. The absence of the element means that no Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party.'

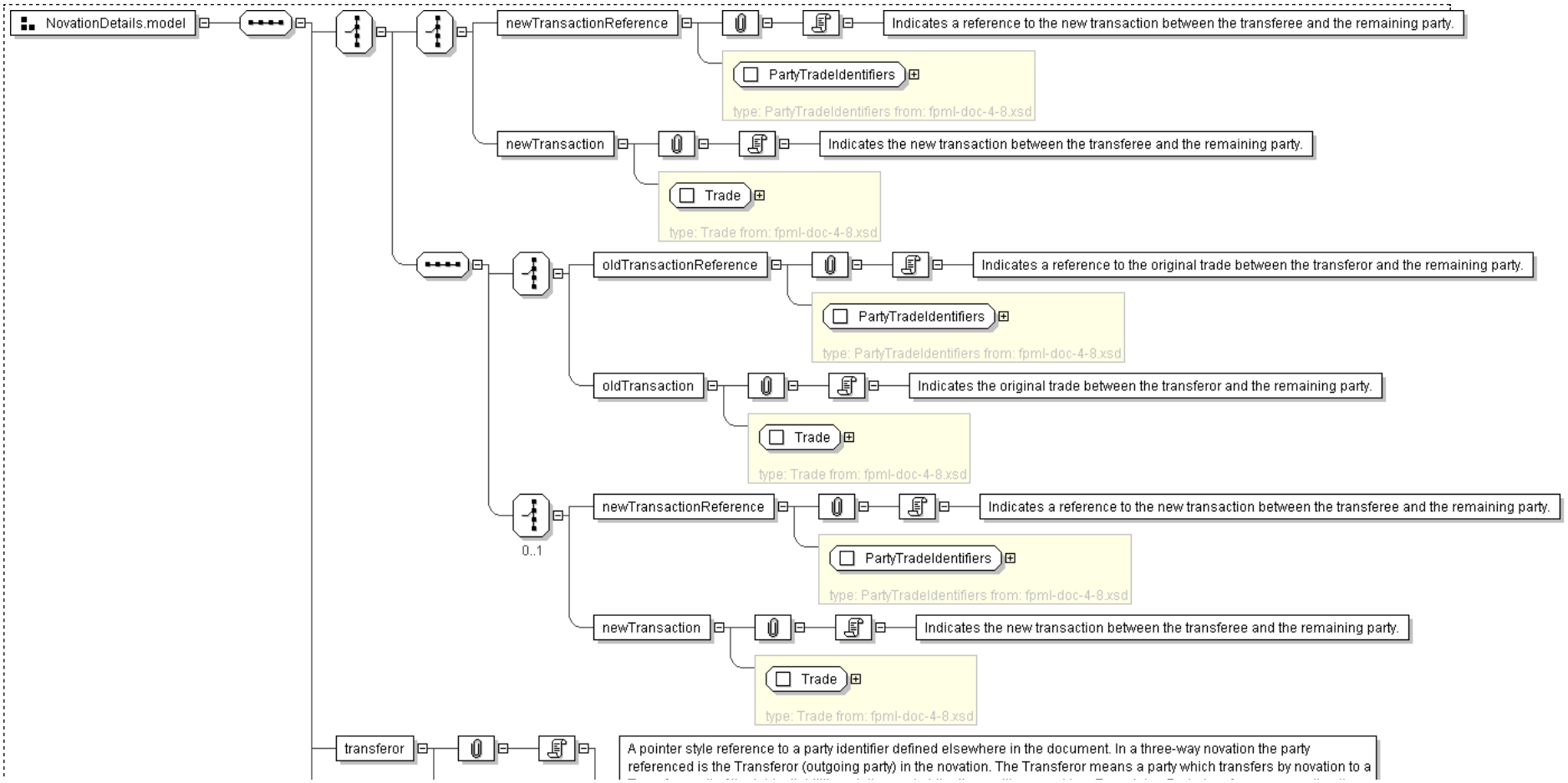
<contractualDefinitions> ContractualDefinitions </contractualDefinitions> [0..*]
'The definitions (such as those published by ISDA) that will define the terms of the novation transaction.'

Start Choice [1]
<contractualSupplement> ContractualSupplement </contractualSupplement> [0..*]
'DEPRECATED - This element will be removed in the next major version of FpML. The element contractualTermsSupplement should be used instead. Definition: A contractual supplement (such as those published by ISDA) that will apply to the trade.'

<contractualTermsSupplement> ContractualTermsSupplement </contractualTermsSupplement> [0..*]
'A contractual supplement (such as those published by ISDA) that will apply to the trade.'

End Choice

Diagram



Transferee all of its rights, liabilities, duties and obligations with respect to a Remaining Party. In a four-way novation the party referenced is Transferor 1 which transfers by novation to Transferee 1 all of its rights, liabilities, duties and obligations with respect to Transferor 2. ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

transferee

A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferee (incoming party) in the novation. Transferee means a party which accepts by way of novation all rights, liabilities, duties and obligations of a Transferor with respect to a Remaining Party. In a four-way novation the party referenced is Transferee 1 which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

remainingParty

A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the novation. Remaining Party means a party which consents to a Transferor's transfer by novation and the acceptance thereof by the Transferee of all of the Transferor's rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation) or Transferor 2 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

otherRemainingParty
0..1

A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).

PartyReference
type: PartyReference from: fpml-shared-4-8.xsd

novationDate

Specifies the date that one party's legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of the 2004 ISDA Novation Definitions, section 1.16.

date
type: date from: XMLSchema.xsd

novationTradeDate
0..1

Specifies the date the parties agree to assign or novate a trade. If this element is not specified, the novationTradeDate will be deemed to be the novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.

date
type: date from: XMLSchema.xsd

0..1

novatedAmount

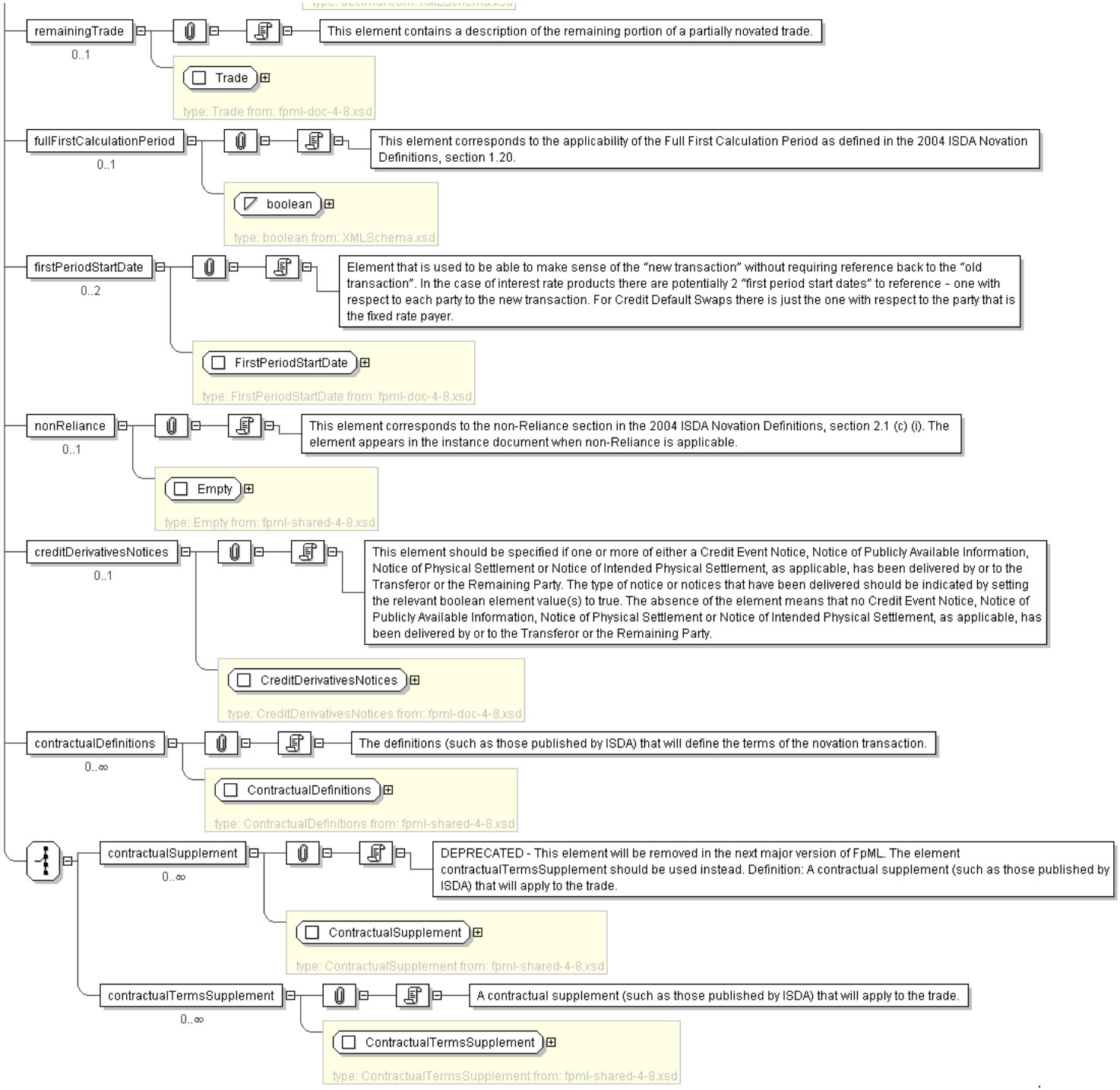
The amount which represents the portion of the Old Transaction being novated.

Money
type: Money from: fpml-shared-4-8.xsd

novatedNumberOfOptions

The number of options which represent the portion of the Old Transaction being novated.

decimal
type: decimal from: XMLSchema.xsd



Schema Component Representation

```
<xsd:group name="NovationDetails.model">
  <xsd:sequence>
```

```
      <xsd:choice>
         <xsd:choice>
            <xsd:element name="newTransactionReference" type=" PartyTradeIdentifiers "/>
            <xsd:element name="newTransaction" type=" Trade "/>
         </xsd:choice>
         <xsd:sequence>
            <xsd:choice>
               <xsd:element name="oldTransactionReference" type=" PartyTradeIdentifiers "/>
               <xsd:element name="oldTransaction" type=" Trade "/>
            </xsd:choice>
            <xsd:choice minOccurs="0">
               <xsd:element name="newTransactionReference" type=" PartyTradeIdentifiers "/>
               <xsd:element name="newTransaction" type=" Trade "/>
            </xsd:choice>
         </xsd:sequence>
      </xsd:choice>
      <xsd:element name="transferor" type=" PartyReference "/>
      <xsd:element name="transferee" type=" PartyReference "/>
      <xsd:element name="remainingParty" type=" PartyReference "/>
      <xsd:element name="otherRemainingParty" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="novationDate" type=" xsd:date "/>
      <xsd:element name="novationTradeDate" type=" xsd:date " minOccurs="0"/>
      <xsd:choice minOccurs="0">
         <xsd:element name="novatedAmount" type=" Money "/>
         <xsd:element name="novatedNumberOfOptions" type=" xsd:decimal "/>
      </xsd:choice>
      <xsd:element name="remainingTrade" type=" Trade " minOccurs="0"/>
      <xsd:element name="fullFirstCalculationPeriod" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="firstPeriodStartDate" type=" FirstPeriodStartDate " minOccurs="0" maxOccurs="2"/>
      <xsd:element name="nonReliance" type=" Empty " minOccurs="0"/>
      <xsd:element name="creditDerivativesNotices" type=" CreditDerivativesNotices " minOccurs="0"/>
      <xsd:element name="contractualDefinitions" type=" ContractualDefinitions " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:choice>
         <xsd:element name="contractualSupplement" type=" ContractualSupplement " minOccurs="0" maxOccurs="unbounded" deprecated="true" deprecatedReason="The
         contractualTermsSupplement includes the publication date, which was not present in the contractualSupplement"/>
         <xsd:element name="contractualTermsSupplement" type=" ContractualTermsSupplement " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:choice>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: NovationMessage.model

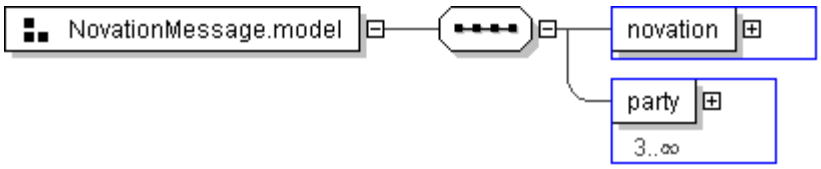
[Table of contents]

Name	NovationMessage.model
Used by (from the same schema document)	Complex Type NovationNotificationMessage , Complex Type NovationRequestMessage , Complex Type NovationResponseMessage

XML Instance Representation

```
<novation> Novation </novation> [1]
<party> Party </party> [3..*]
```

Diagram



Schema Component Representation

```
<xsd:group name="NovationMessage.model">
  <xsd:sequence>
    <xsd:element name="novation" type=" Novation "/>
    <xsd:element name="party" type=" Party " minOccurs="3" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: TerminationDetails.model

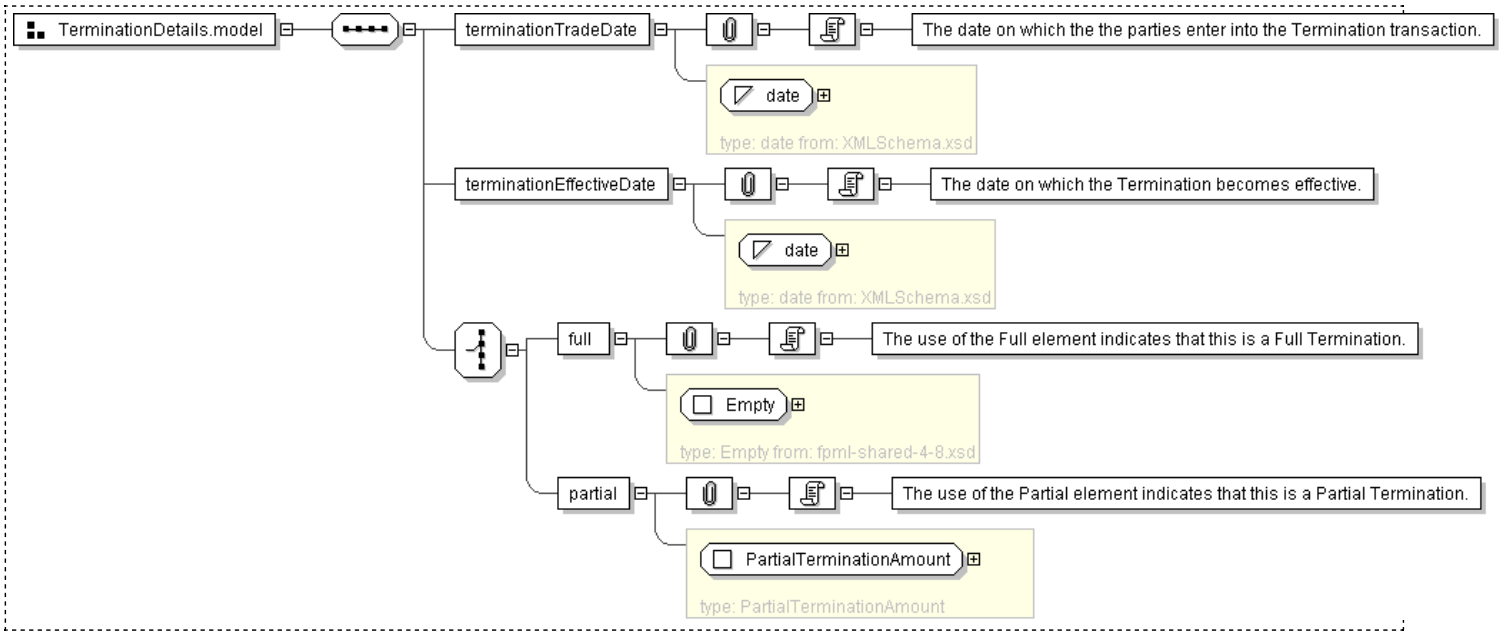
[Table of contents]

Name	TerminationDetails.model
Used by (from the same schema document)	Complex Type Termination

XML Instance Representation

```
<terminationTradeDate> xsd:date </terminationTradeDate> [1]
'The date on which the the parties enter into the Termination transaction.'

<terminationEffectiveDate> xsd:date </terminationEffectiveDate> [1]
'The date on which the Termination becomes effective.'

Start Choice [1]
   <full> Empty </full> [1]
   'The use of the Full element indicates that this is a Full Termination.'

   <partial> PartialTerminationAmount </partial> [1]
   'The use of the Partial element indicates that this is a Partial Termination.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="TerminationDetails.model">
   <xsd:sequence>
      <xsd:element name="terminationTradeDate" type=" xsd:date "/>
      <xsd:element name="terminationEffectiveDate" type=" xsd:date "/>
      <xsd:choice>
         <xsd:element name="full" type=" Empty "/>
         <xsd:element name="partial" type=" PartialTerminationAmount "/>
      </xsd:choice>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AffectedTransactions

[Table of contents]

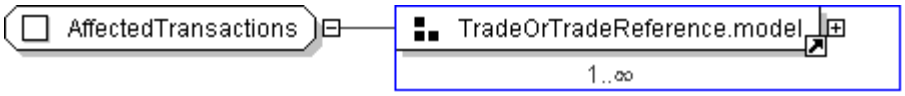
Super-types:	None
Sub-types:	None

Name	AffectedTransactions
Abstract	no

XML Instance Representation

```
<...>
  Start Group: TradeOrTradeReference.model [1..*]
    Start Choice [1]
      <trade> Trade </trade> [1]
        'An element that allows the full details of the trade to be used as a
        mechanism for identifying the trade for which the post-trade event
        pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
        'A container since an individual trade can be referenced by two or more
        different partyTradeIdentifier elements - each allocated by a different
        party.'

    End Choice
  End Group: TradeOrTradeReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AffectedTransactions">
  <xsd:group ref=" TradeOrTradeReference.model " maxOccurs="unbounded"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Novation

[Table of contents]

Super-types:	Event < Novation (by extension)
Sub-types:	None

Name	Novation
Used by (from the same schema document)	Model Group NovationMessage.model
Abstract	no
Documentation	An event type that records the occurrence of a novation

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

   Start Choice [1]
      Start Choice [1]
         <newTransactionReference> PartyTradeIdentifiers </newTransactionReference> [1]
         'Indicates a reference to the new transaction between the transferee and
         the remaining party.'

         <newTransaction> Trade </newTransaction> [1]
         'Indicates the new transaction between the transferee and the remaining
         party.'

      End Choice
      Start Choice [1]
         <oldTransactionReference> PartyTradeIdentifiers </oldTransactionReference> [1]
         'Indicates a reference to the original trade between the transferor and
         the remaining party.'

         <oldTransaction> Trade </oldTransaction> [1]
         'Indicates the original trade between the transferor and the remaining
         party.'

      End Choice
      Start Choice [0..1]
         <newTransactionReference> PartyTradeIdentifiers </newTransactionReference> [1]
         'Indicates a reference to the new transaction between the transferee and
         the remaining party.'

         <newTransaction> Trade </newTransaction> [1]
         'Indicates the new transaction between the transferee and the remaining
         party.'

      End Choice
   End Choice
   <transferor> PartyReference </transferor> [1]
   'A pointer style reference to a party identifier defined elsewhere in the
   document. In a three-way novation the party referenced is the Transferor
   (outgoing party) in the novation. The Transferor means a party which transfers
   by novation to a Transferee all of its rights, liabilities, duties and
   obligations with respect to a Remaining Party. In a four-way novation the party
   referenced is Transferor 1 which transfers by novation to Transferee 1 all of
   its rights, liabilities, duties and obligations with respect to Transferor 2.
   ISDA 2004 Novation Term: Transferor (three-way novation) or Transferor 1 (four-
```

way novation).'

<transferee> PartyReference **</transferee>** [1]

'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Transferee (incoming party) in the novation. Transferee means a party which accepts by way of novation all rights, liabilities, duties and obligations of a Transferor with respect to a Remaining Party. In a four-way novation the party referenced is Transferee 1 which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 1. ISDA 2004 Novation Term: Transferee (three-way novation) or Transferee 1 (four-way novation).'

<remainingParty> PartyReference **</remainingParty>** [1]

'A pointer style reference to a party identifier defined elsewhere in the document. In a three-way novation the party referenced is the Remaining Party in the novation. Remaining Party means a party which consents to a Transferor\'s transfer by novation and the acceptance thereof by the Transferee of all of the Transferor\'s rights, liabilities, duties and obligations with respect to such Remaining Party under and with respect of the Novated Amount of a transaction. In a four-way novation the party referenced is Transferor 2 per the ISDA definition and acts in the role of a Transferor. Transferor 2 transfers by novation to Transferee 2 all of its rights, liabilities, duties and obligations with respect to Transferor 1. ISDA 2004 Novation Term: Remaining Party (three-way novation) or Transferor 2 (four-way novation).'

<otherRemainingParty> PartyReference **</otherRemainingParty>** [0..1]

'A pointer style reference to a party identifier defined elsewhere in the document. This element is not applicable in a three-way novation and should be omitted. In a four-way novation the party referenced is Transferee 2. Transferee 2 means a party which accepts by way of novation the rights, liabilities, duties and obligations of Transferor 2. ISDA 2004 Novation Term: Transferee 2 (four-way novation).'

<novationDate> xsd:date **</novationDate>** [1]

'Specifies the date that one party\'s legal obligations with regard to a trade are transferred to another party. It corresponds to the Novation Date section of the 2004 ISDA Novation Definitions, section 1.16.'

<novationTradeDate> xsd:date **</novationTradeDate>** [0..1]

'Specifies the date the parties agree to assign or novate a trade. If this element is not specified, the novationTradeDate will be deemed to be the novationDate. It corresponds to the Novation Trade Date section of the 2004 ISDA Novation Definitions, section 1.17.'

Start Choice [0..1]

<novatedAmount> Money **</novatedAmount>** [1]

'The amount which represents the portion of the Old Transaction being novated.'

<novatedNumberOfOptions> xsd:decimal **</novatedNumberOfOptions>** [1]

'The number of options which represent the portion of the Old Transaction being novated.'

End Choice

<remainingTrade> Trade **</remainingTrade>** [0..1]

'This element contains a description of the remaining portion of a partially novated trade.'

<fullFirstCalculationPeriod> xsd:boolean **</fullFirstCalculationPeriod>** [0..1]

'This element corresponds to the applicability of the Full First Calculation Period as defined in the 2004 ISDA Novation Definitions, section 1.20.'

<firstPeriodStartDate> FirstPeriodStartDate **</firstPeriodStartDate>** [0..2]

'Element that is used to be able to make sense of the â€œnew transactionâ€ without requiring reference back to the â€œold transactionâ€. In the case of interest rate products there are potentially 2 â€œfirst period start datesâ€ to reference â€“ one with respect to each party to the new transaction. For Credit Default Swaps there is just the one with respect to the party that is the fixed rate payer.'

<nonReliance> Empty **</nonReliance>** [0..1]

'This element corresponds to the non-Reliance section in the 2004 ISDA Novation Definitions, section 2.1 (c) (i). The element appears in the instance document when non-Reliance is applicable.'

<creditDerivativesNotices> CreditDerivativesNotices **</creditDerivativesNotices>** [0..1]

'This element should be specified if one or more of either a Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party. The type of notice or notices that have been delivered should be indicated by setting the relevant boolean element value(s) to true. The absence of the element means that no Credit Event Notice, Notice of Publicly Available Information, Notice of Physical Settlement or Notice of Intended Physical Settlement, as applicable, has been delivered by or to the Transferor or the Remaining Party.'

<contractualDefinitions> ContractualDefinitions **</contractualDefinitions>** [0..*]

'The definitions (such as those published by ISDA) that will define the terms of the novation transaction.'

Start Choice [1]

<contractualSupplement> ContractualSupplement **</contractualSupplement>** [0..*]

'DEPRECATED - This element will be removed in the next major version of FpML. The element contractualTermsSupplement should be used instead. Definition: A contractual supplement (such as those published by ISDA) that will apply to the trade.'

<contractualTermsSupplement> ContractualTermsSupplement **</contractualTermsSupplement>** [0..*]

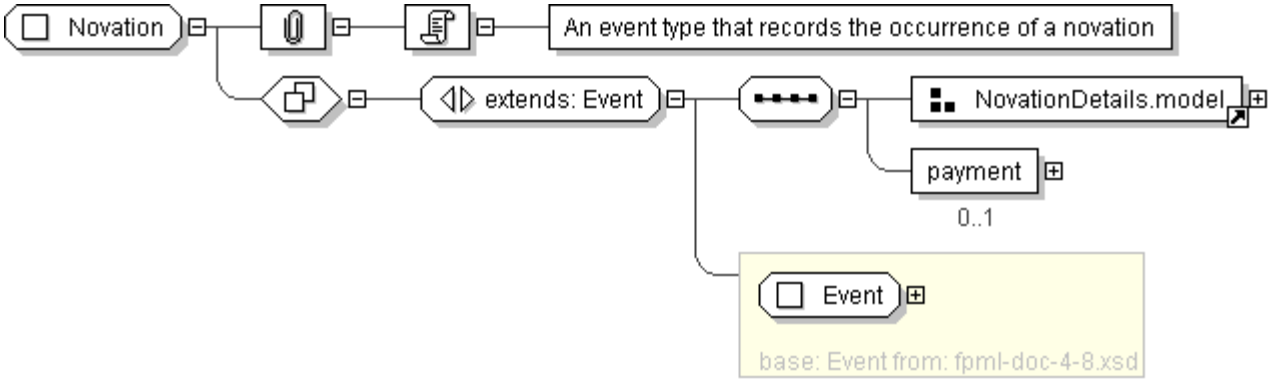
'A contractual supplement (such as those published by ISDA) that will apply to the trade.'

End Choice

<payment> Payment **</payment>** [0..1]

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="Novation">
  <xsd:complexContent>
    <xsd:extension base=" Event ">
      <xsd:sequence>
        <xsd:group ref=" NovationDetails.model "/>
        <xsd:element name="payment" type=" Payment " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationNotificationMessage

[Table of contents]

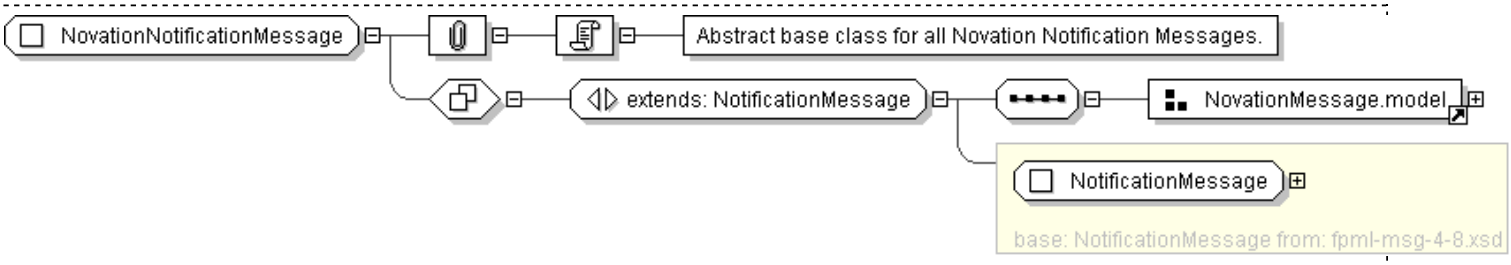
Super-types:	NotificationMessage < NovationNotificationMessage (by extension)
Sub-types:	None

Name	NovationNotificationMessage
Abstract	yes
Documentation	Abstract base class for all Novation Notification Messages.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationNotificationMessage" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:group ref=" NovationMessage.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationRequestMessage

[Table of contents]

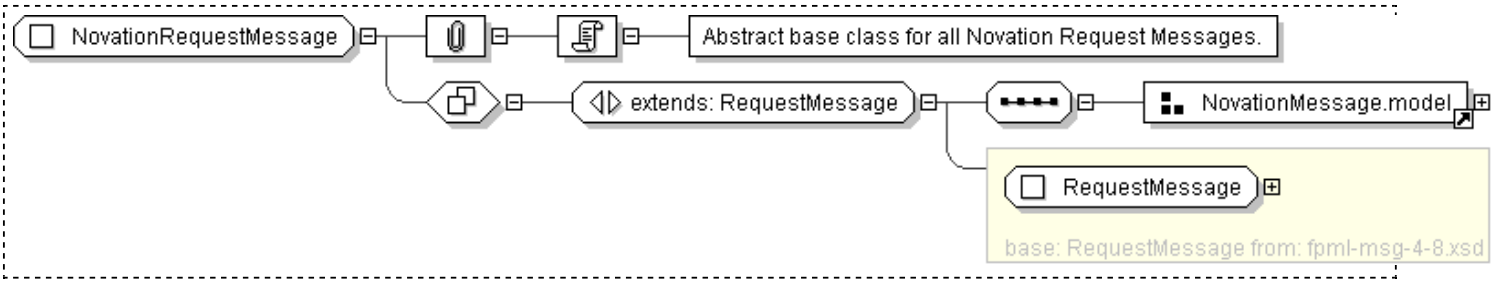
Super-types:	RequestMessage < NovationRequestMessage (by extension)
Sub-types:	None

Name	NovationRequestMessage
Abstract	yes
Documentation	Abstract base class for all Novation Request Messages.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationRequestMessage" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" RequestMessage ">
         <xsd:sequence>
            <xsd:group ref=" NovationMessage.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationResponseMessage

[Table of contents]

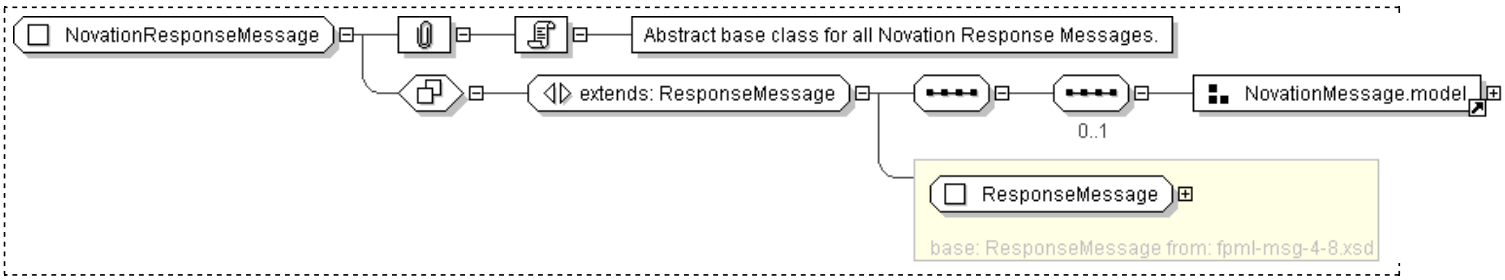
Super-types:	ResponseMessage < NovationResponseMessage (by extension)
Sub-types:	None

Name	NovationResponseMessage
Abstract	yes
Documentation	Abstract base class for all Novation Response Messages.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Sequence [0..1]
      <novation> Novation </novation> [1]
      <party> Party </party> [3..*]
   End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationResponseMessage" abstract="true">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:sequence minOccurs="0">
          <xsd:group ref=" NovationMessage.model "/>
        </xsd:sequence>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartialTerminationAmount

[Table of contents]

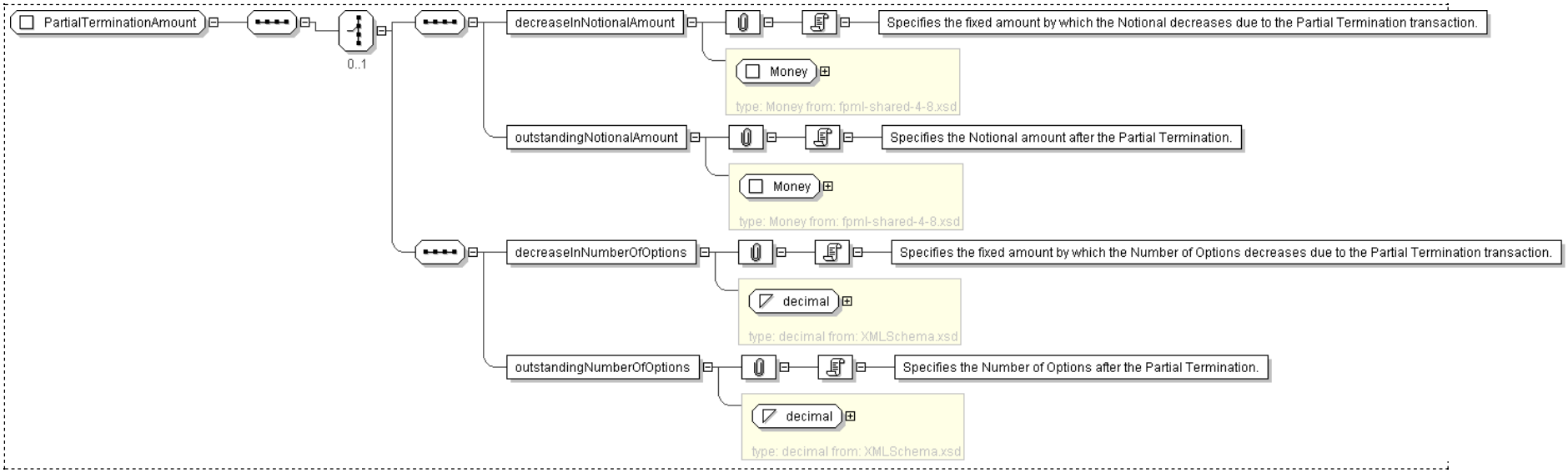
Super-types:	None
Sub-types:	None

Name	PartialTerminationAmount
Used by (from the same schema document)	Model Group TerminationDetails.model
Abstract	no

XML Instance Representation

```
<...>
  Start Choice [0..1]
    <decreaseInNotionalAmount> Money </decreaseInNotionalAmount> [1]
    'Specifies the fixed amount by which the Notional decreases due to the Partial Termination transaction.'

    <outstandingNotionalAmount> Money </outstandingNotionalAmount> [1]
    'Specifies the Notional amount after the Partial Termination.'

    <decreaseInNumberOfOptions> xsd:decimal </decreaseInNumberOfOptions> [1]
    'Specifies the fixed amount by which the Number of Options decreases due to the Partial Termination transaction.'

    <outstandingNumberOfOptions> xsd:decimal </outstandingNumberOfOptions> [1]
    'Specifies the Number of Options after the Partial Termination.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartialTerminationAmount">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:sequence>
        <xsd:element name="decreaseInNotionalAmount" type=" Money "/>
        <xsd:element name="outstandingNotionalAmount" type=" Money "/>
      </xsd:sequence>
      <xsd:sequence>
        <xsd:element name="decreaseInNumberOfOptions" type=" xsd:decimal "/>
        <xsd:element name="outstandingNumberOfOptions" type=" xsd:decimal "/>
      </xsd:sequence>
    </xsd:choice>
```

```
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Termination

[Table of contents]

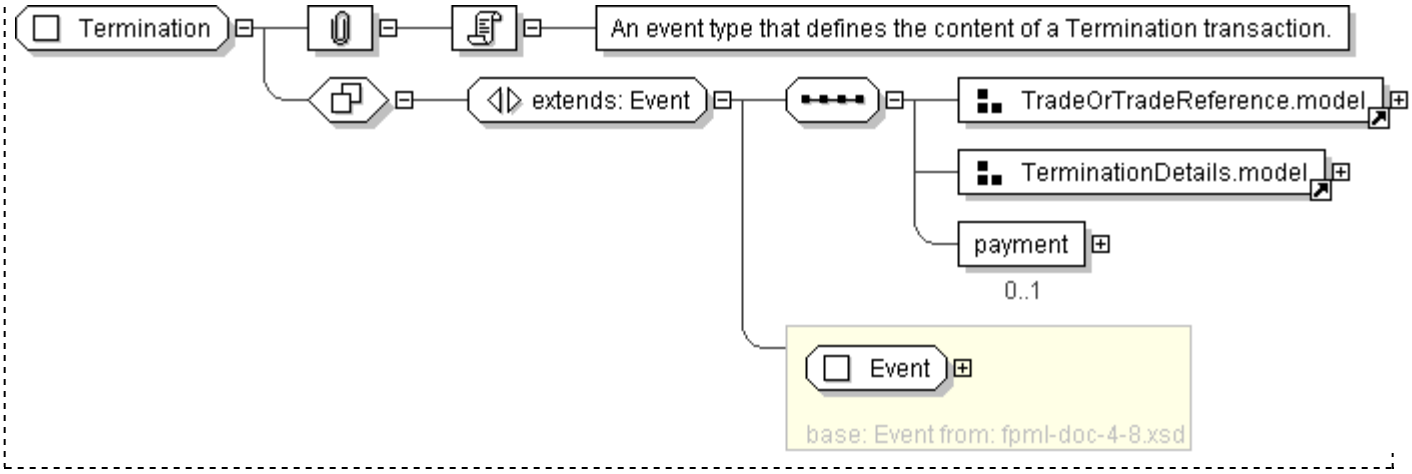
Super-types:	Event < Termination (by extension)
Sub-types:	None

Name	Termination
Abstract	no
Documentation	An event type that defines the content of a Termination transaction.

XML Instance Representation

```
<...>
   <eventId> EventId </eventId> [0..*]
   ''

   Start Choice [1]
      <trade> Trade </trade> [1]
      'An element that allows the full details of the trade to be used as a
      mechanism for identifying the trade for which the post-trade event pertains'

      <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
      'A container since an individual trade can be referenced by two or more
      different partyTradeIdentifier elements - each allocated by a different
      party.'

   End Choice
   <terminationTradeDate> xsd:date </terminationTradeDate> [1]
   'The date on which the the parties enter into the Termination transaction.'

   <terminationEffectiveDate> xsd:date </terminationEffectiveDate> [1]
   'The date on which the Termination becomes effective.'

   Start Choice [1]
      <full> Empty </full> [1]
      'The use of the Full element indicates that this is a Full Termination.'

      <partial> PartialTerminationAmount </partial> [1]
      'The use of the Partial element indicates that this is a Partial
      Termination.'

   End Choice
   <payment> Payment </payment> [0..1]
   'A payment for the right to terminate the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Termination">
   <xsd:complexContent>
      <xsd:extension base=" Event ">
         <xsd:sequence>
            <xsd:group ref=" TradeOrTradeReference.model "/>
            <xsd:group ref=" TerminationDetails.model "/>
            <xsd:element name="payment" type=" Payment " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAmendment

[Table of contents]

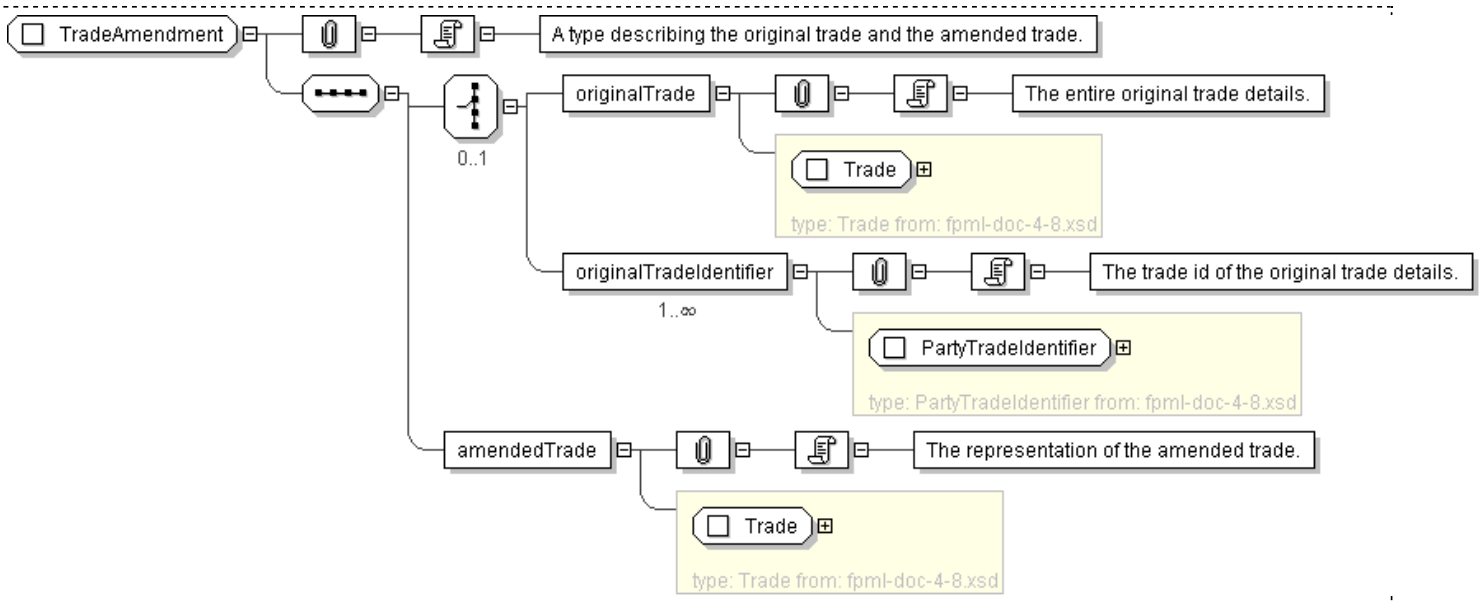
Super-types:	None
Sub-types:	None

Name	TradeAmendment
Abstract	no
Documentation	A type describing the original trade and the amended trade.

XML Instance Representation

```
<...>
  Start Choice [0..1]
    <originalTrade> Trade </originalTrade> [1]
     'The entire original trade details.'

    <originalTradeIdentifier> PartyTradeIdentifier </originalTradeIdentifier> [1..*]
     'The trade id of the original trade details.'

  End Choice
  <amendedTrade> Trade </amendedTrade> [1]
  'The representation of the amended trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAmendment">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="originalTrade" type=" Trade "/>
      <xsd:element name="originalTradeIdentifier" type=" PartyTradeIdentifier " maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:element name="amendedTrade" type=" Trade "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: AmendmentConfirmed
  - Complex Type: IncreaseConfirmed
  - Complex Type: NovationAlleged
  - Complex Type: NovationConfirmed
  - Complex Type: RequestAmendmentConfirmation
  - Complex Type: RequestIncreaseConfirmation
  - Complex Type: RequestNovationConfirmation
  - Complex Type: RequestTerminationConfirmation
  - Complex Type: TerminationConfirmed
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
```

```
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AmendmentConfirmed

[Table of contents]

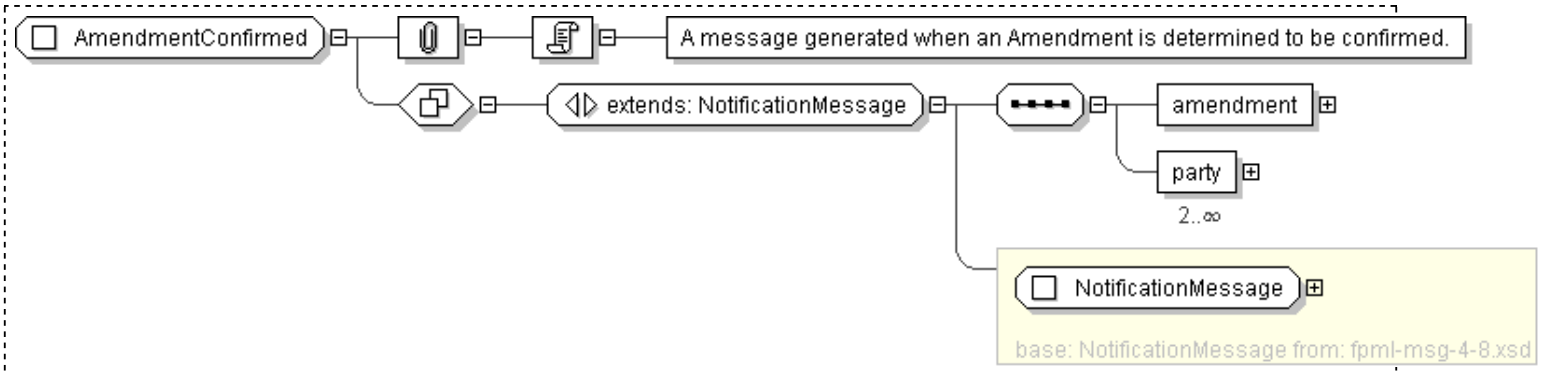
Super-types:	NotificationMessage < AmendmentConfirmed (by extension)
Sub-types:	None

Name	AmendmentConfirmed
Abstract	no
Documentation	A message generated when an Amendment is determined to be confirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> Amendment </amendment> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AmendmentConfirmed">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
```

```
            <xsd:element name="amendment" type=" Amendment "/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IncreaseConfirmed

[Table of contents]

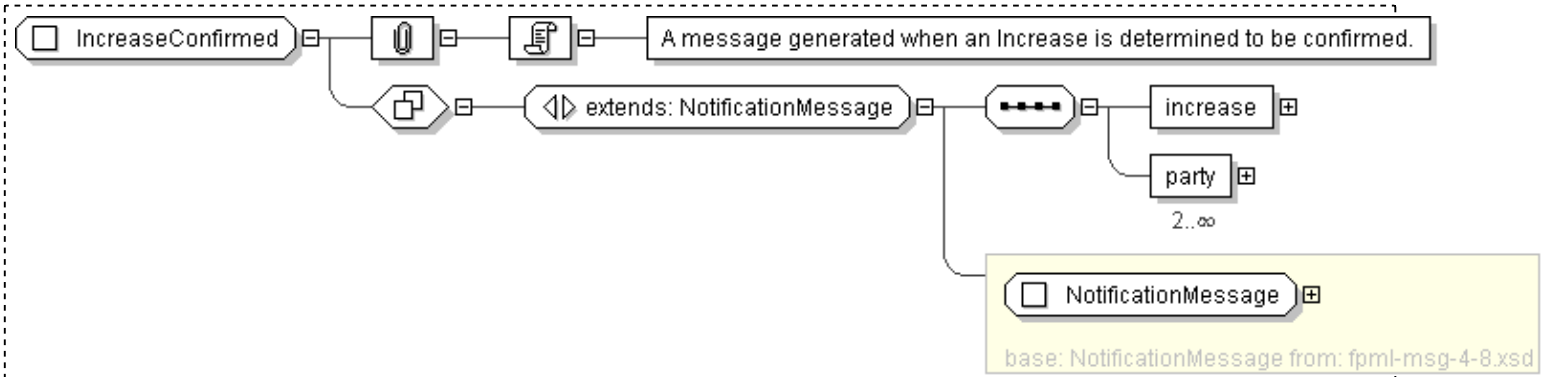
Super-types:	NotificationMessage < IncreaseConfirmed (by extension)
Sub-types:	None

Name	IncreaseConfirmed
Abstract	no
Documentation	A message generated when an Increase is determined to be confirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <increase> Increase </increase> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IncreaseConfirmed">
  <xsd:complexContent>
```

```
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:element name="increase" type=" Increase "/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationAlleged

[Table of contents]

Super-types:	NovationNotificationMessage < NovationAlleged (by extension)
Sub-types:	None

Name	NovationAlleged
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="NovationAlleged">
   <xsd:complexContent>
      <xsd:extension base=" NovationNotificationMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: NovationConfirmed

[Table of contents]

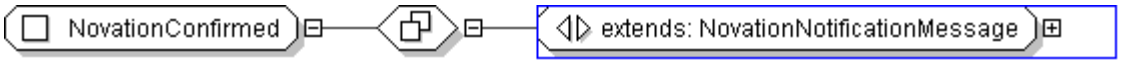
Super-types:	NovationNotificationMessage < NovationConfirmed (by extension)
Sub-types:	None

Name	NovationConfirmed
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationConfirmed">
   <xsd:complexContent>
      <xsd:extension base=" NovationNotificationMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestAmendmentConfirmation

[Table of contents]

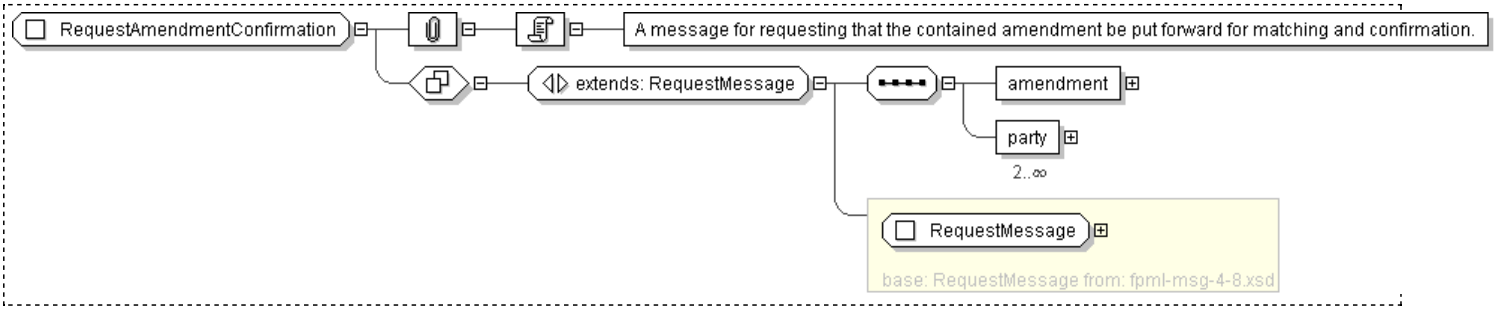
Super-types:	RequestMessage < RequestAmendmentConfirmation (by extension)
Sub-types:	None

Name	RequestAmendmentConfirmation
Abstract	no
Documentation	A message for requesting that the contained amendment be put forward for matching and confirmation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> Amendment </amendment> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestAmendmentConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="amendment" type=" Amendment "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestIncreaseConfirmation

[Table of contents]

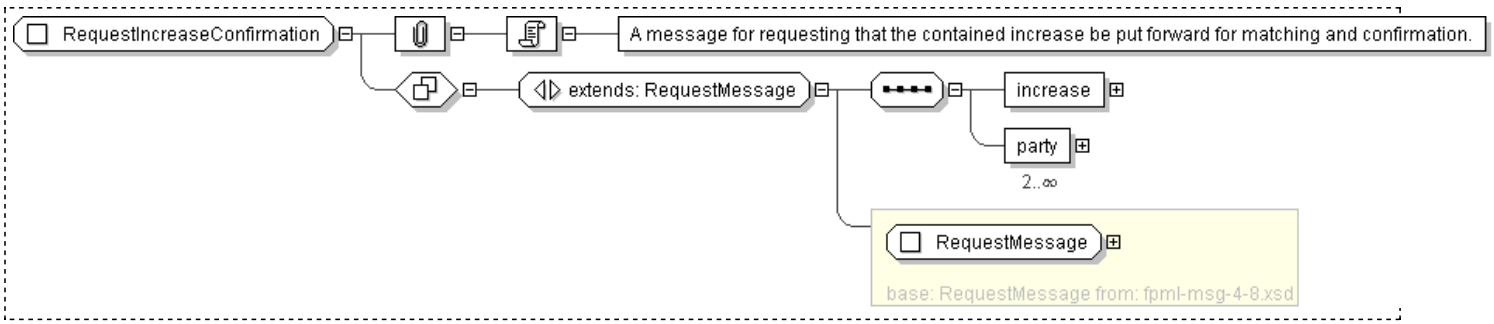
Super-types:	RequestMessage < RequestIncreaseConfirmation (by extension)
Sub-types:	None

Name	RequestIncreaseConfirmation
Abstract	no
Documentation	A message for requesting that the contained increase be put forward for matching and confirmation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no
changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation)
the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <increase> Increase </increase> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestIncreaseConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="increase" type=" Increase "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestNovationConfirmation

[Table of contents]

Super-types:	NovationRequestMessage < RequestNovationConfirmation (by extension)
Sub-types:	None

Name	RequestNovationConfirmation
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="RequestNovationConfirmation">
   <xsd:complexContent>
      <xsd:extension base=" NovationRequestMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestTerminationConfirmation

[Table of contents]

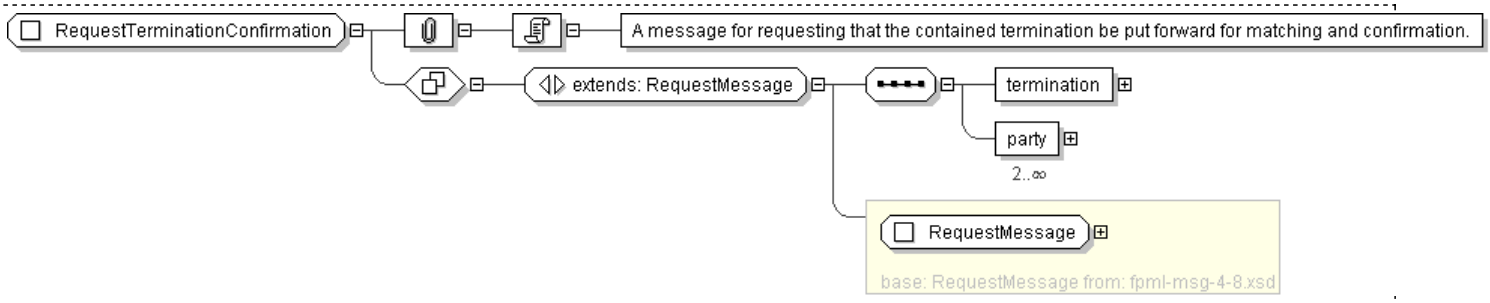
Super-types:	RequestMessage < RequestTerminationConfirmation (by extension)
Sub-types:	None

Name	RequestTerminationConfirmation
Abstract	no
Documentation	A message for requesting that the contained termination be put forward for matching and confirmation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> Termination </termination> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestTerminationConfirmation">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="termination" type=" Termination "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TerminationConfirmed

[Table of contents]

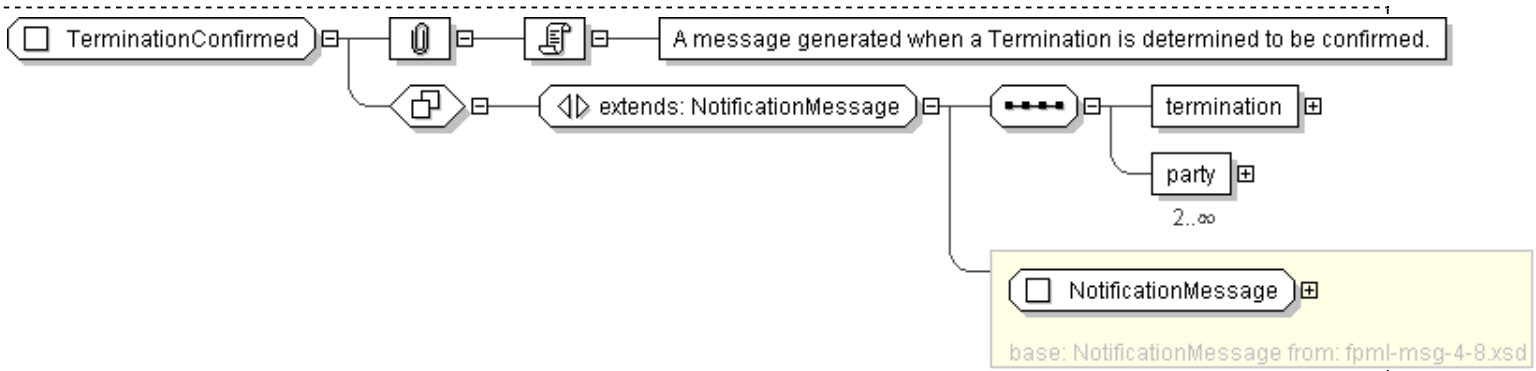
Super-types:	NotificationMessage < TerminationConfirmed (by extension)
Sub-types:	None

Name	TerminationConfirmed
Abstract	no
Documentation	A message generated when a Termination is determined to be confirmed.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> Termination </termination> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TerminationConfirmed">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
```

```
            <xsd:element name="termination" type=" Termination "/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: NovateTrade
  - Complex Type: TradeNovated
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovateTrade

[Table of contents]

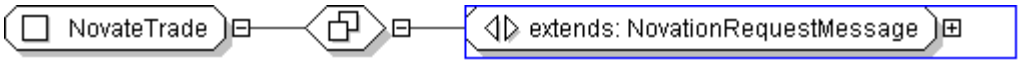
Super-types:	NovationRequestMessage < NovateTrade (by extension)
Sub-types:	None

Name	NovateTrade
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovateTrade">
   <xsd:complexContent>
      <xsd:extension base=" NovationRequestMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeNovated

[Table of contents]

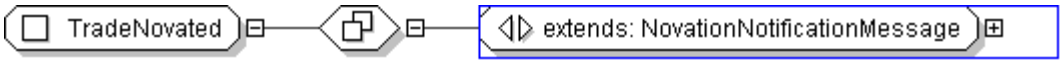
Super-types:	NovationNotificationMessage < TradeNovated (by extension)
Sub-types:	None

Name	TradeNovated
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeNovated">
   <xsd:complexContent>
      <xsd:extension base=" NovationNotificationMessage "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: NovationConsentGranted
  - Complex Type: NovationConsentRefused
  - Complex Type: NovationConsentRequest
  - Complex Type: TradeAmendmentRequest
  - Complex Type: TradeAmendmentResponse
  - Complex Type: TradeIncreaseRequest
  - Complex Type: TradeIncreaseResponse
  - Complex Type: TradeTerminationRequest
  - Complex Type: TradeTerminationResponse
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-posttrade-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-posttrade-4-8.xsd"/>
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
```

```
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationConsentGranted

[Table of contents]

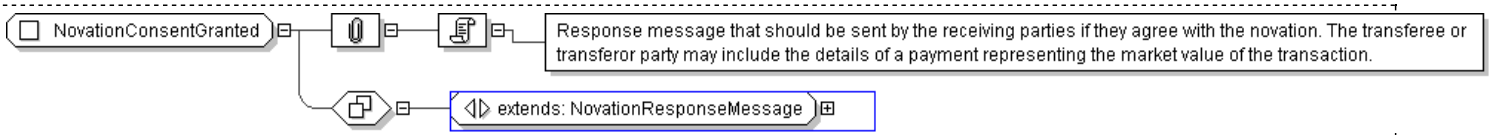
Super-types:	NovationResponseMessage < NovationConsentGranted (by extension)
Sub-types:	None

Name	NovationConsentGranted
Abstract	no
Documentation	Response message that should be sent by the receiving parties if they agree with the novation. The transferee or transferor party may include the details of a payment representing the market value of the transaction.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number
stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Sequence [0..1]
      <novation> Novation </novation> [1]
      <party> Party </party> [3..*]
   End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationConsentGranted">
  <xsd:complexContent>
    <xsd:extension base=" NovationResponseMessage "/>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationConsentRefused

[Table of contents]

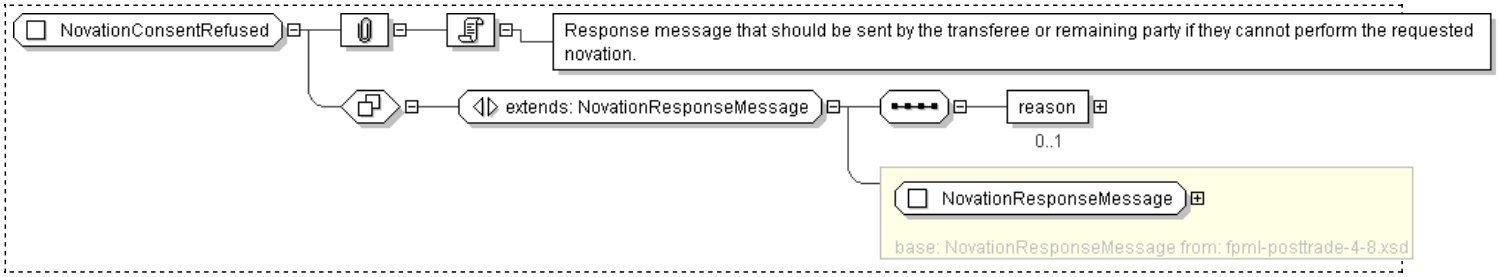
Super-types:	NovationResponseMessage < NovationConsentRefused (by extension)
Sub-types:	None

Name	NovationConsentRefused
Abstract	no
Documentation	Response message that should be sent by the transferee or remaining party if they cannot perform the requested novation.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the
build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples
within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between
releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Sequence [0..1]
      <novation> Novation </novation> [1]
      <party> Party </party> [3..*]
   End Sequence
   <reason> Reason </reason> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationConsentRefused">
  <xsd:complexContent>
    <xsd:extension base=" NovationResponseMessage ">
      <xsd:sequence>
        <xsd:element name="reason" type=" Reason " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NovationConsentRequest

[Table of contents]

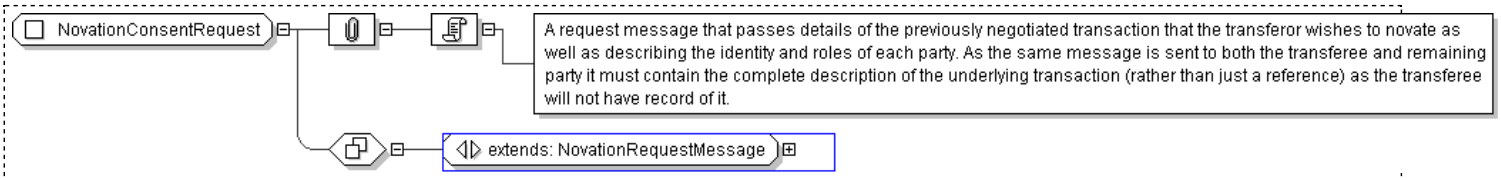
Super-types:	NovationRequestMessage < NovationConsentRequest (by extension)
Sub-types:	None

Name	NovationConsentRequest
Abstract	no
Documentation	A request message that passes details of the previously negotiated transaction that the transferor wishes to novate as well as describing the identity and roles of each party. As the same message is sent to both the transferee and remaining party it must contain the complete description of the underlying transaction (rather than just a reference) as the transferee will not have record of it.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the
schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it
is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of
the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version
(e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a
version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <novation> Novation </novation> [1]
   <party> Party </party> [3..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NovationConsentRequest">
  <xsd:complexContent>
    <xsd:extension base=" NovationRequestMessage "/>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAmendmentRequest

[Table of contents]

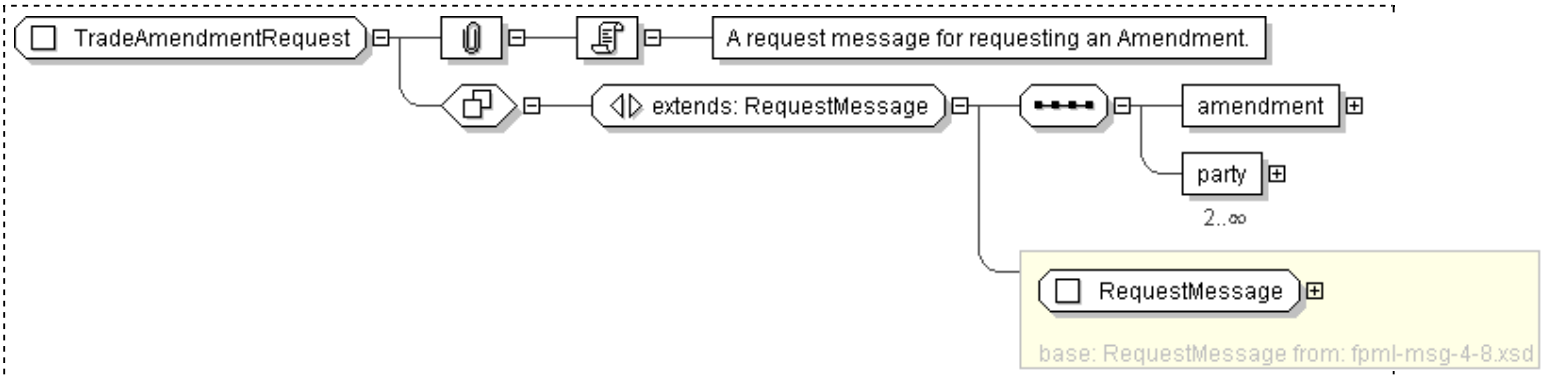
Super-types:	RequestMessage < TradeAmendmentRequest (by extension)
Sub-types:	None

Name	TradeAmendmentRequest
Abstract	no
Documentation	A request message for requesting an Amendment.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> Amendment </amendment> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAmendmentRequest">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
```

```
        <xsd:sequence>
           <xsd:element name="amendment" type=" Amendment "/>
           <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
     </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAmendmentResponse

[Table of contents]

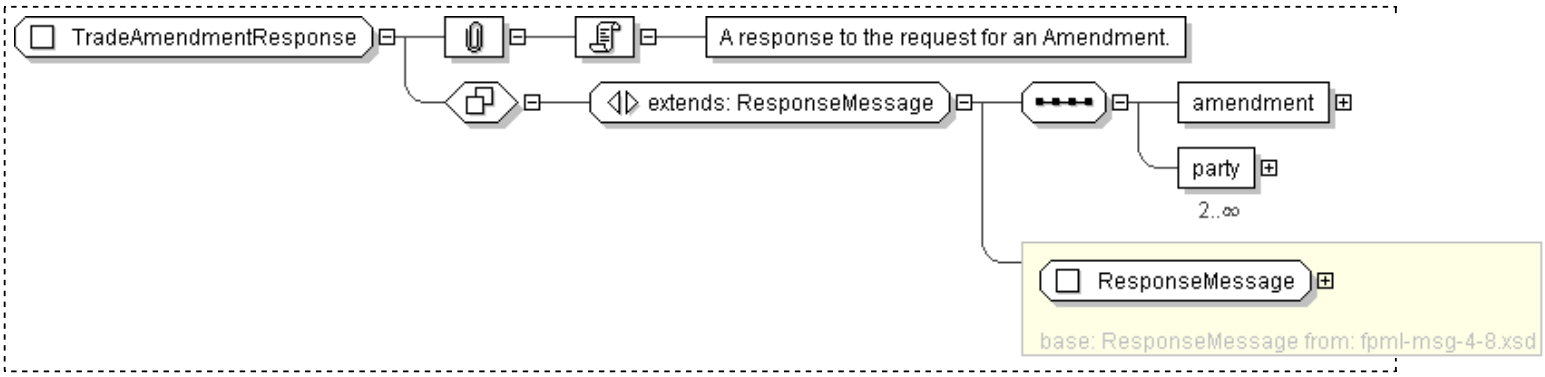
Super-types:	ResponseMessage < TradeAmendmentResponse (by extension)
Sub-types:	None

Name	TradeAmendmentResponse
Abstract	no
Documentation	A response to the request for an Amendment.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <amendment> Amendment </amendment> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAmendmentResponse">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="amendment" type=" Amendment "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeIncreaseRequest

[Table of contents]

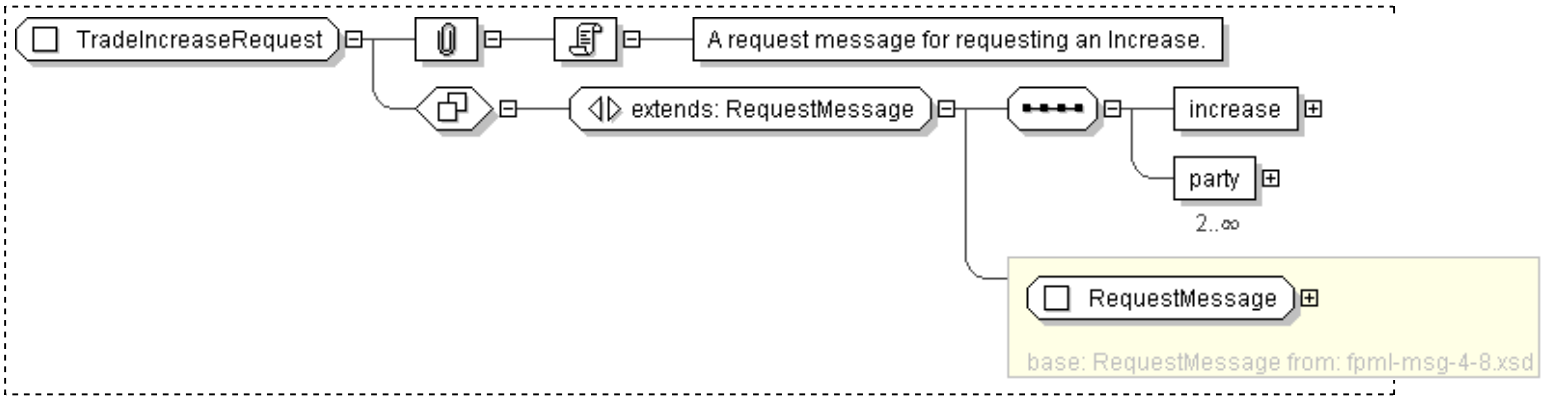
Super-types:	RequestMessage < TradeIncreaseRequest (by extension)
Sub-types:	None

Name	TradeIncreaseRequest
Abstract	no
Documentation	A request message for requesting an Increase.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <increase> Increase </increase> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeIncreaseRequest">
```

```
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="increase" type=" Increase "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeIncreaseResponse

[Table of contents]

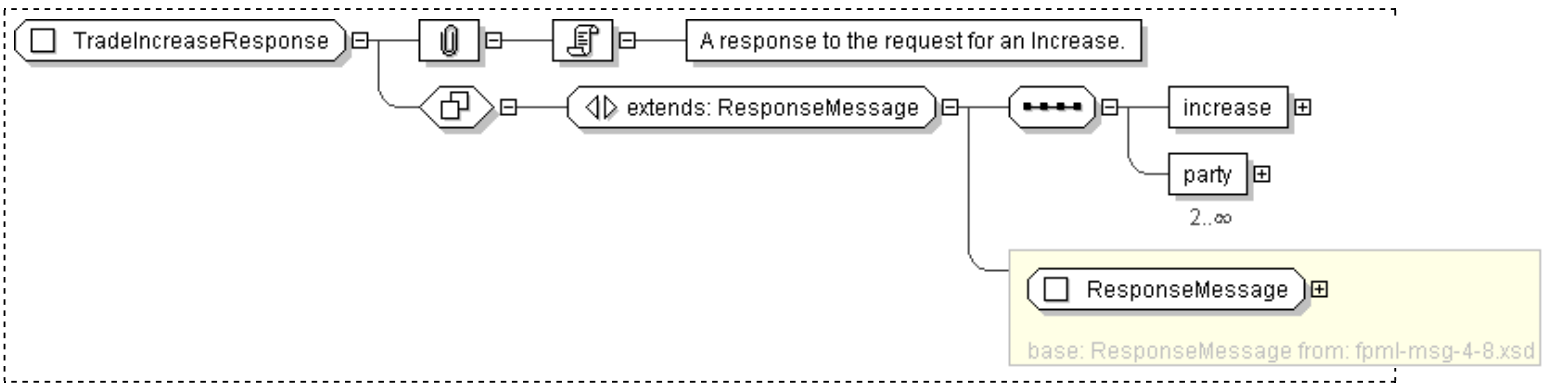
Super-types:	ResponseMessage < TradeIncreaseResponse (by extension)
Sub-types:	None

Name	TradeIncreaseResponse
Abstract	no
Documentation	A response to the request for an Increase.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <increase> Increase </increase> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeIncreaseResponse">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="        " type=" ________ "/>
```

```
                                increase          Increase
          <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeTerminationRequest

[Table of contents]

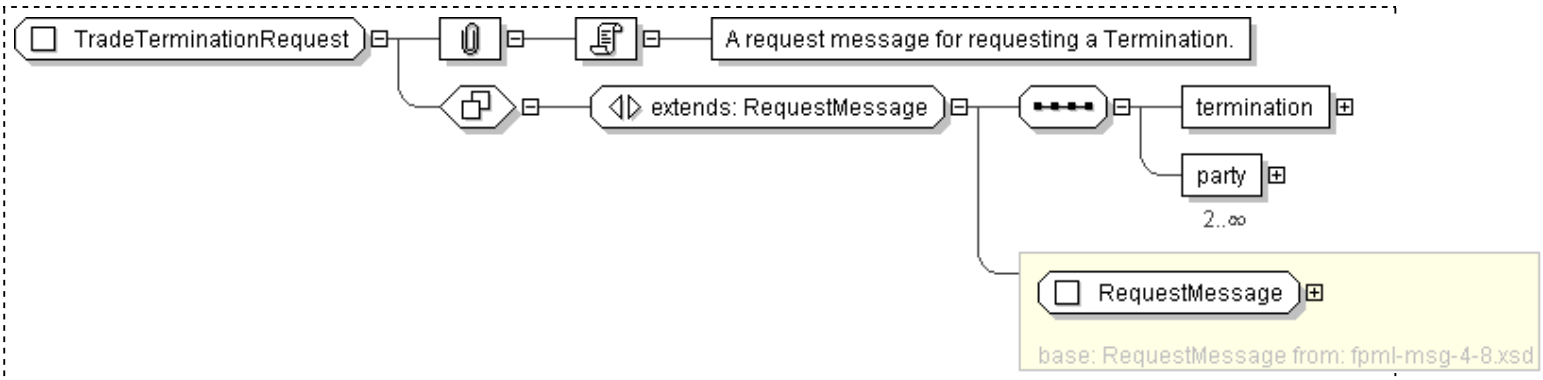
Super-types:	RequestMessage < TradeTerminationRequest (by extension)
Sub-types:	None

Name	TradeTerminationRequest
Abstract	no
Documentation	A request message for requesting a Termination.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> Termination </termination> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeTerminationRequest">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
```

```
        <xsd:sequence>
           <xsd:element name="termination" type=" Termination "/>
           <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
     </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeTerminationResponse

[Table of contents]

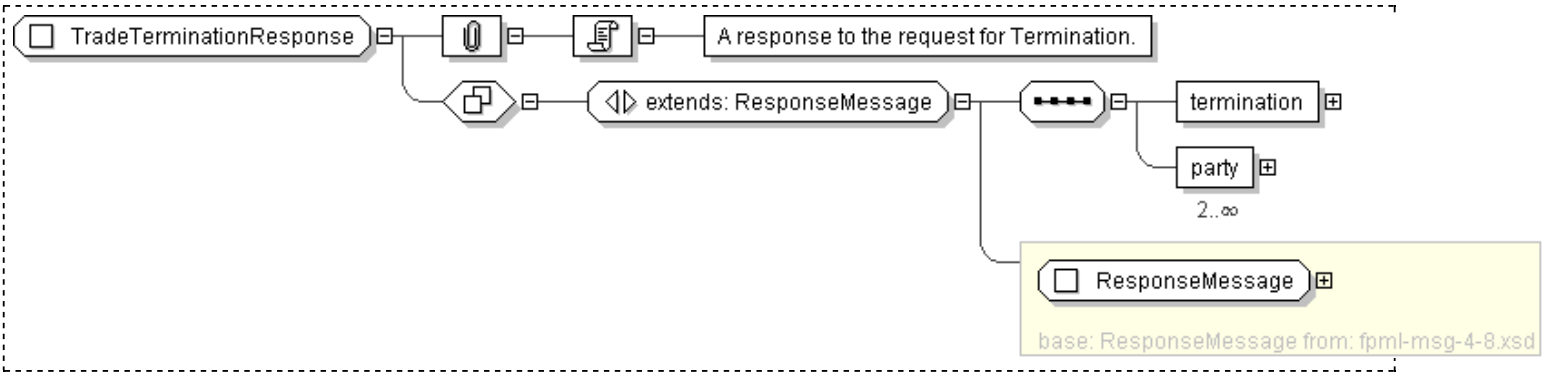
Super-types:	ResponseMessage < TradeTerminationResponse (by extension)
Sub-types:	None

Name	TradeTerminationResponse
Abstract	no
Documentation	A response to the request for Termination.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <termination> Termination </termination> [1]
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeTerminationResponse">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="termination" type=" Termination "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: quotableFxSingleLeg
  - Element: quotableProduct
- Global Definitions
  - Complex Type: AcceptQuote
  - Complex Type: QuotableFxLeg
  - Complex Type: QuotableFxRate
  - Complex Type: QuotablePayment
  - Complex Type: QuotableProduct
  - Complex Type: Quote
  - Complex Type: QuoteAcceptanceConfirmed
  - Complex Type: QuoteAlreadyExpired
  - Complex Type: QuoteUpdated
  - Complex Type: RequestQuote
  - Complex Type: RequestQuoteResponse
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml-annotation	http://www.fpml.org/annotation
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
```

```
            <simpleType>
              <restriction base=" string ">
                <pattern value="[1-9][0-9]{3}"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character

32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

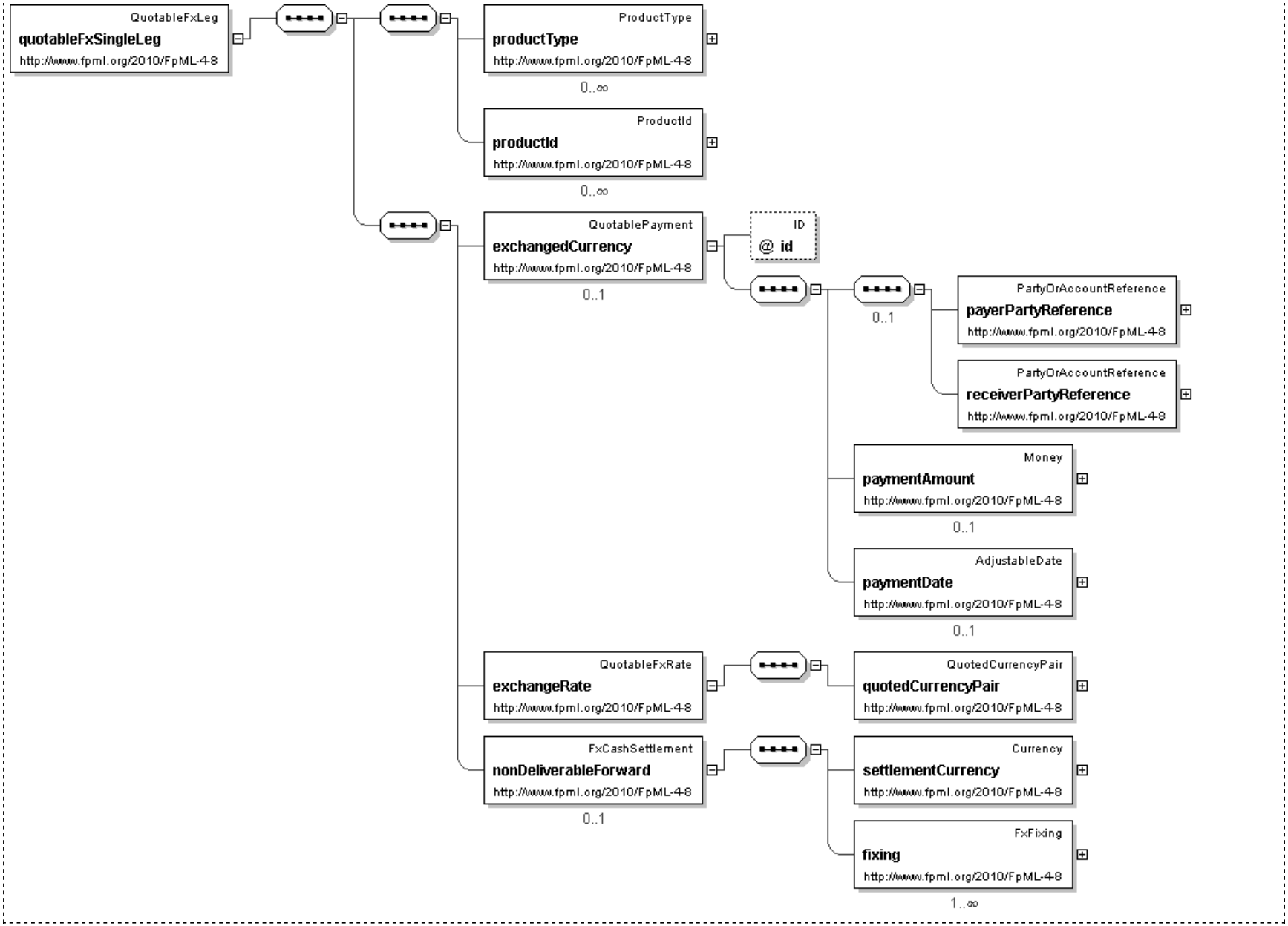
Element: quotableFxSingleLeg

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - quotableProduct

Name	quotableFxSingleLeg
Type	QuotableFxLeg
Nillable	no
Abstract	no

Logical Diagram

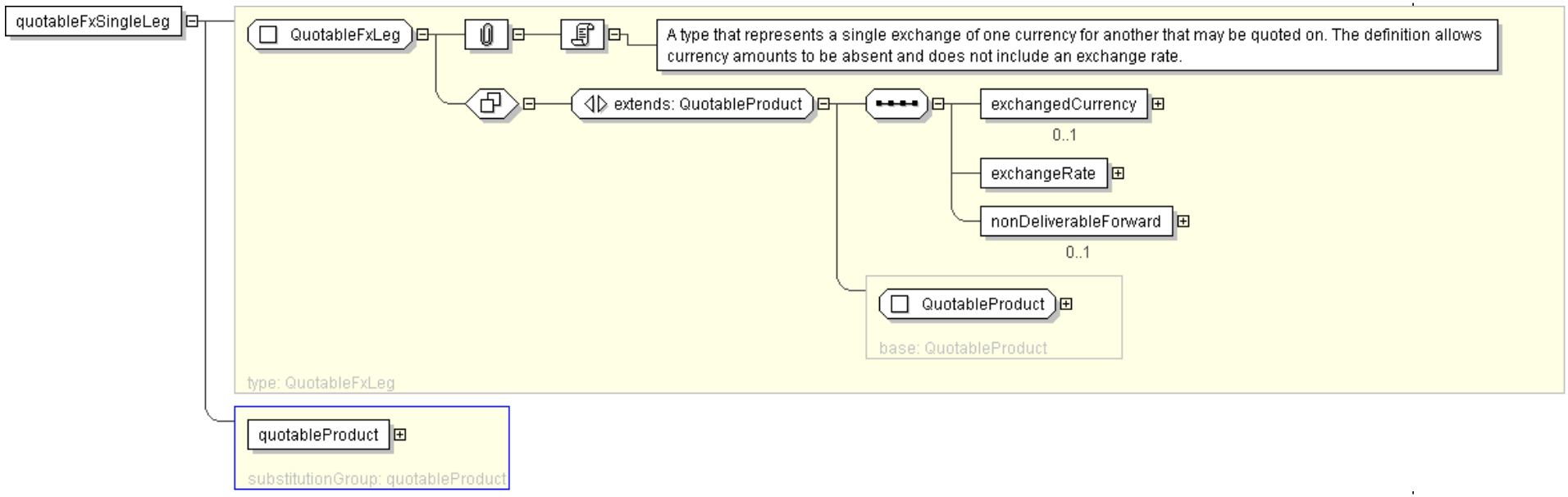


XML Instance Representation

```
<quotableFxSingleLeg>
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element.
   Note that the domain values for this element are not strictly an enumerated list.'

   <exchangedCurrency> QuotablePayment </exchangedCurrency> [0..1]
   <exchangeRate> QuotableFxRate </exchangeRate> [1]
   <nonDeliverableForward> FxCashSettlement </nonDeliverableForward> [0..1]
```

```
    'Used to describe a particular type of FX forward transaction that is settled in a single currency.'
</quotableFxSingleLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="quotableFxSingleLeg" type=" QuotableFxLeg " substitutionGroup="quotableProduct"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

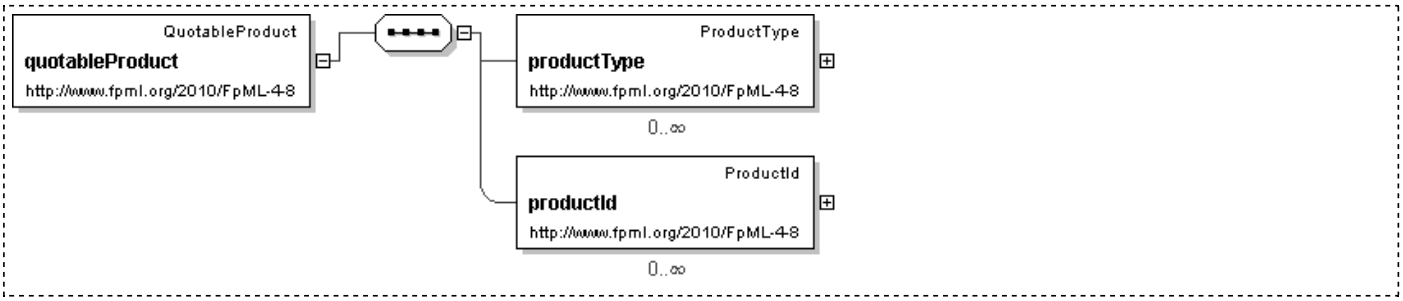
Element: quotableProduct

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - quotableFxSingleLeg

Name	quotableProduct
Used by (from the same schema document)	Complex Type QuoteUpdated , Complex Type RequestQuote
Type	QuotableProduct
Nillable	no
Abstract	yes

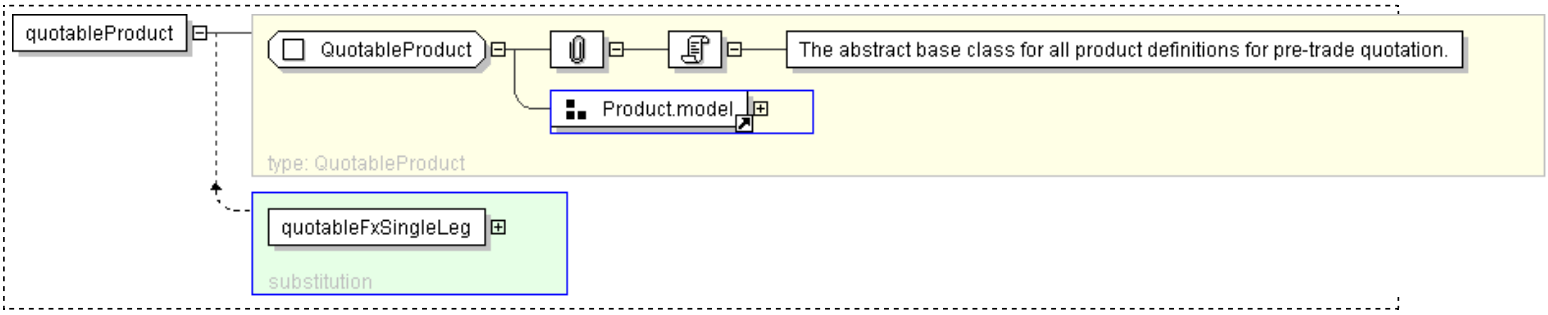
Logical Diagram



XML Instance Representation

```
<quotableProduct>
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a
    coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values
    associated with this element. Note that the domain values for this element are not strictly an
    enumerated list.'

</quotableProduct>
```

Diagram



Schema Component Representation

```
<xsd:element name="quotableProduct" type=" QuotableProduct " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AcceptQuote

[Table of contents]

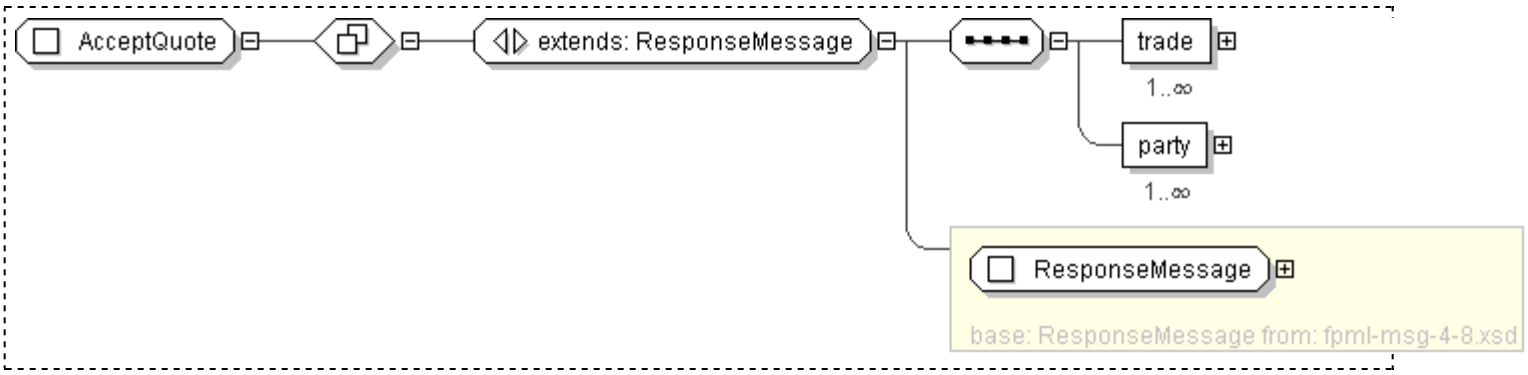
Super-types:	ResponseMessage < AcceptQuote (by extension)
Sub-types:	None

Name	AcceptQuote
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   trade> Trade </trade> [1..*]
   party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AcceptQuote">
   <xsd:complexContent>
      <xsd:extension base=" ResponseMessage ">
         <xsd:sequence>
```

```
          <xsd:element name="trade" type=" Trade " maxOccurs="unbounded"/>
          <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotableFxLeg

[Table of contents]

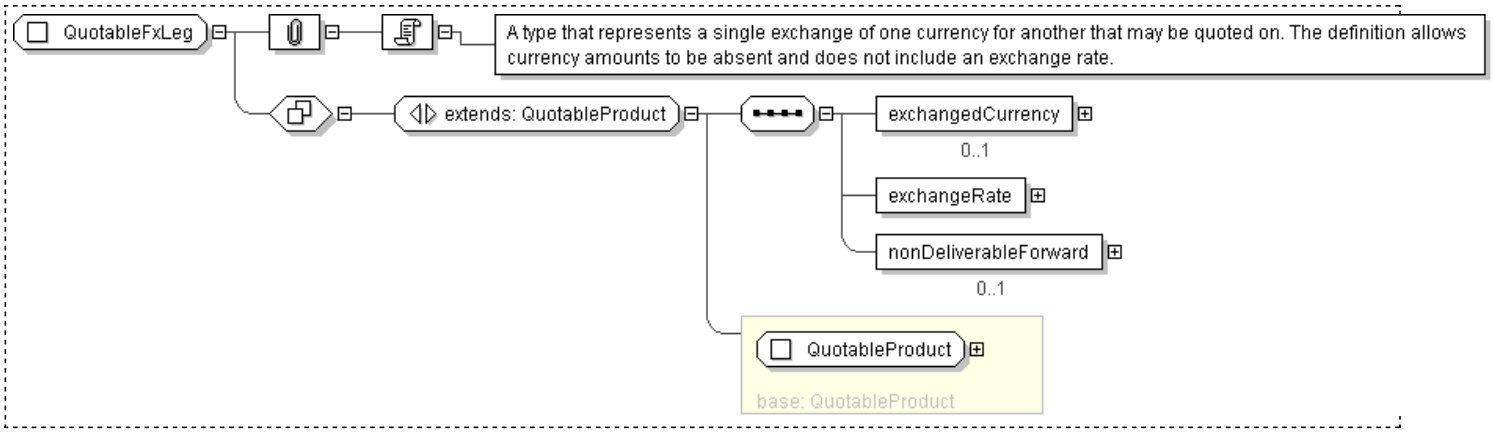
Super-types:	QuotableProduct < QuotableFxLeg (by extension)
Sub-types:	None

Name	QuotableFxLeg
Used by (from the same schema document)	Element quotableFxSingleLeg
Abstract	no
Documentation	A type that represents a single exchange of one currency for another that may be quoted on. The definition allows currency amounts to be absent and does not include an exchange rate.

XML Instance Representation

```
<...>
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated
   with this element. Note that the domain values for this element are not strictly an enumerated list.'

   <exchangedCurrency> QuotablePayment </exchangedCurrency> [0..1]
   <exchangeRate> QuotableFxRate </exchangeRate> [1]
   <nonDeliverableForward> FxCashSettlement </nonDeliverableForward> [0..1]
   'Used to describe a particular type of FX forward transaction that is settled in a single currency.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotableFxLeg">
   <xsd:complexContent>
      <xsd:extension base=" QuotableProduct ">
         <xsd:sequence>
            <xsd:element name="exchangedCurrency" type=" QuotablePayment " minOccurs="0"/>
            <xsd:element name="exchangeRate" type=" QuotableFxRate "/>
            <xsd:element name="nonDeliverableForward" type=" FxCashSettlement " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotableFxRate

[Table of contents]

Super-types:	None
Sub-types:	None

Name	QuotableFxRate
Used by (from the same schema document)	Complex Type QuotableFxLeg
Abstract	no

XML Instance Representation

```
<...>
   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="QuotableFxRate">
   <xsd:sequence>
      <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotablePayment

[Table of contents]

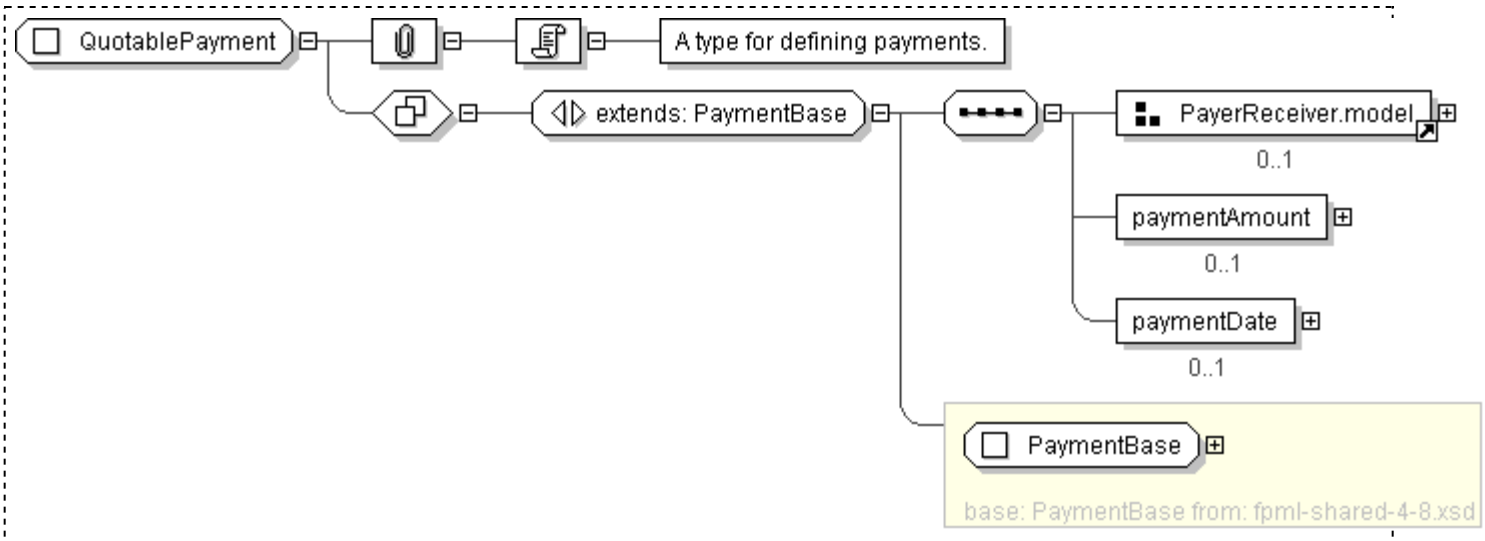
Super-types:	PaymentBase < QuotablePayment (by extension)
Sub-types:	None

Name	QuotablePayment
Used by (from the same schema document)	Complex Type QuotableFxLeg
Abstract	no
Documentation	A type for defining payments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Group: PayerReceiver.model [0..1]
      <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
       'A reference to the party responsible for making the payments defined by this
       structure.'

      <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
       'A reference to the party that receives the payments corresponding to this
       structure.'

   End Group: PayerReceiver.model
   <paymentAmount> Money </paymentAmount> [0..1]
    'The currency amount of the payment.'

   <paymentDate> AdjustableDate </paymentDate> [0..1]
    'The payment date. This date is subject to adjustment in accordance with any
    applicable business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotablePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model " minOccurs="0"/>
            <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
            <xsd:element name="paymentDate" type=" AdjustableDate " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotableProduct

[Table of contents]

Super-types: None

Sub-types:

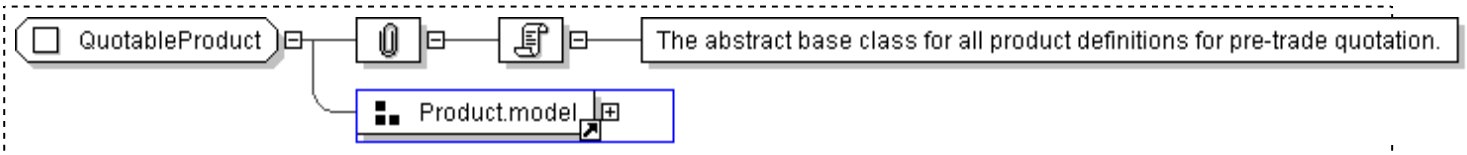
- QuotableFxLeg (by extension)

Name	QuotableProduct
Used by (from the same schema document)	Element quotableProduct
Abstract	yes
Documentation	The abstract base class for all product definitions for pre-trade quotation.

XML Instance Representation

```
<...>
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for this
    element are not strictly an enumerated list.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotableProduct" abstract="true">
   <xsd:group ref=" Product.model "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Quote

[Table of contents]

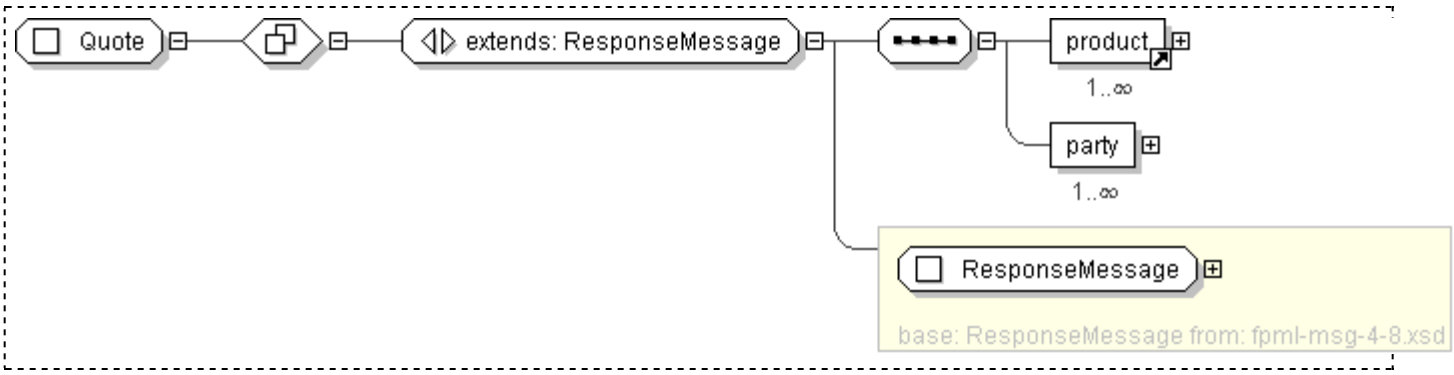
Super-types:	ResponseMessage < Quote (by extension)
Sub-types:	None

Name	Quote
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <product> ... </product> [1..*]
   <party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Quote">
   <xsd:complexContent>
      <xsd:extension base=" ResponseMessage ">
         <xsd:sequence>
```

```
            <xsd:element ref=" product " maxOccurs="unbounded"/>
            <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuoteAcceptanceConfirmed

[Table of contents]

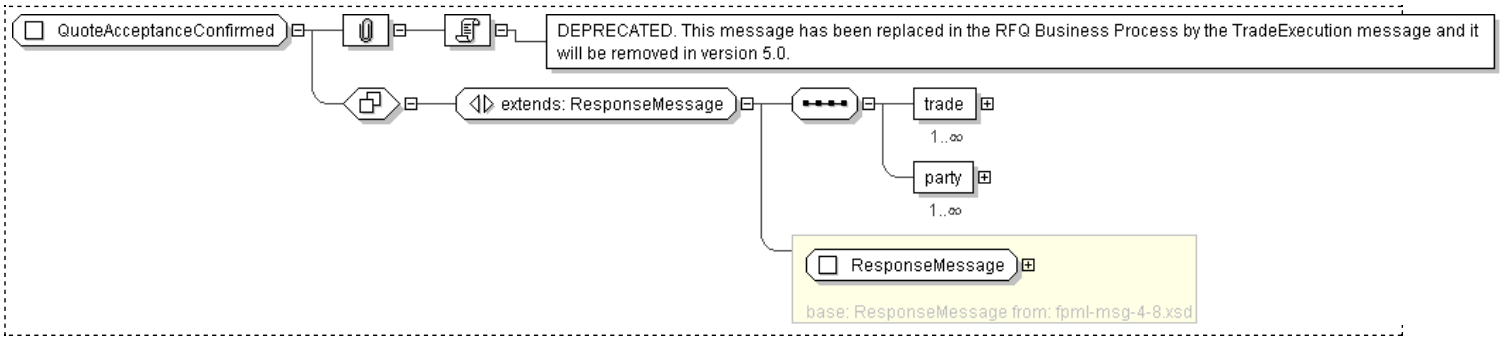
Super-types:	ResponseMessage < QuoteAcceptanceConfirmed (by extension)
Sub-types:	None

Name	QuoteAcceptanceConfirmed
Abstract	no
Documentation	DEPRECATED. This message has been replaced in the RFQ Business Process by the TradeExecution message and it will be removed in version 5.0.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the
schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it
is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of
the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version
(e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a
version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1..*]
   <party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuoteAcceptanceConfirmed" deprecated="true" deprecatedReason="The message has been replaced by the
TradeExecution message to provide the link between the RFQ and Trade Execution processes.">
   <xsd:complexContent>
      <xsd:extension base=" ResponseMessage ">
         <xsd:sequence>
            <xsd:element name="trade" type=" Trade " maxOccurs="unbounded"/>
            <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuoteAlreadyExpired

[Table of contents]

Super-types:	ResponseMessage < QuoteAlreadyExpired (by extension)
Sub-types:	None

Name	QuoteAlreadyExpired
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="QuoteAlreadyExpired">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage "/>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuoteUpdated

[Table of contents]

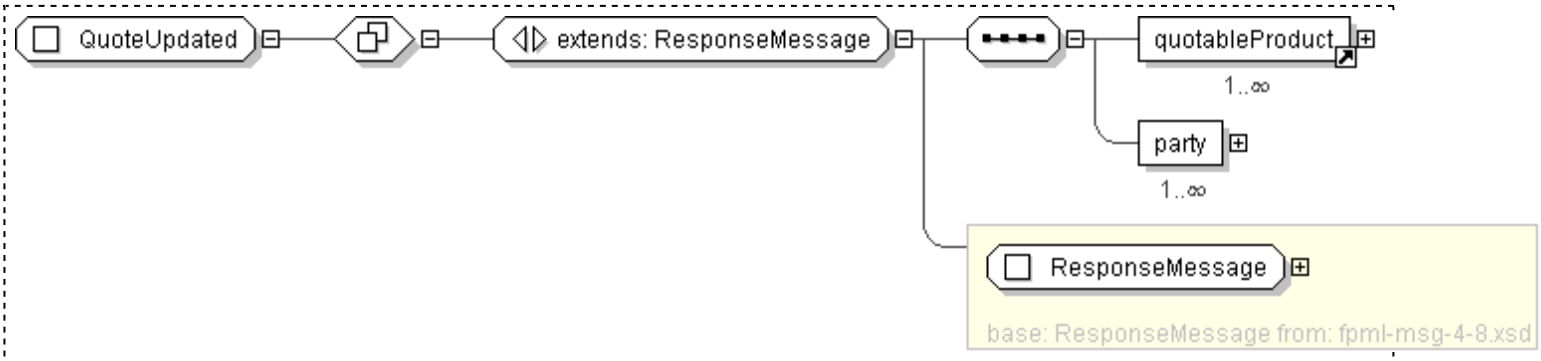
Super-types:	ResponseMessage < QuoteUpdated (by extension)
Sub-types:	None

Name	QuoteUpdated
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <quotableProduct> ... </quotableProduct> [1..*]
   <party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuoteUpdated">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element ref=" quotableProduct " maxOccurs="unbounded"/>
```

```
          <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestQuote

[Table of contents]

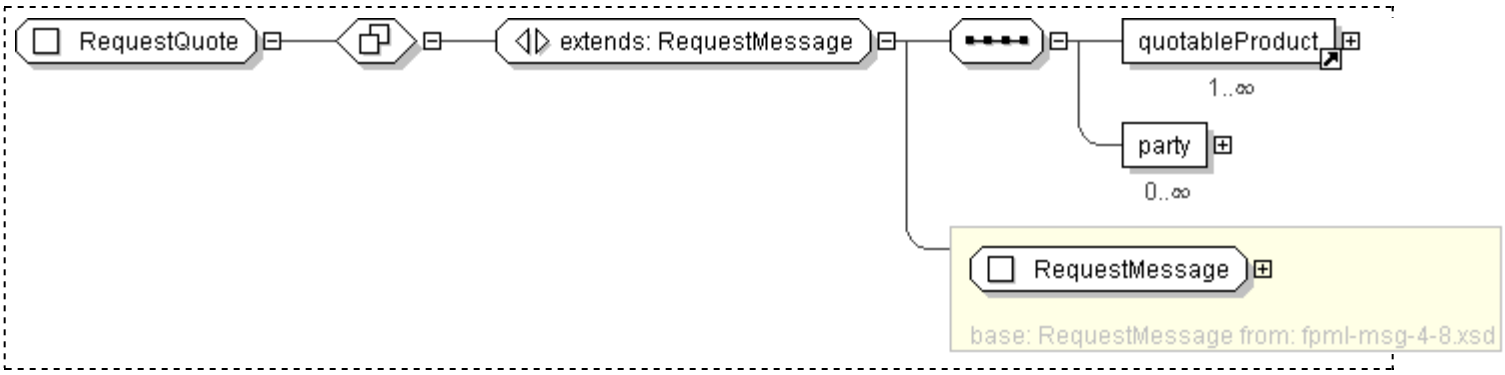
Super-types:	RequestMessage < RequestQuote (by extension)
Sub-types:	None

Name	RequestQuote
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <quotableProduct> ... </quotableProduct> [1..*]
   <party> Party </party> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestQuote">
   <xsd:complexContent>
      <xsd:extension base=" RequestMessage ">
         <xsd:sequence>
```

```
            <xsd:element ref=" quotableProduct " maxOccurs="unbounded"/>
            <xsd:element name="party" type=" Party " minOccurs="0"
            maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestQuoteResponse

[Table of contents]

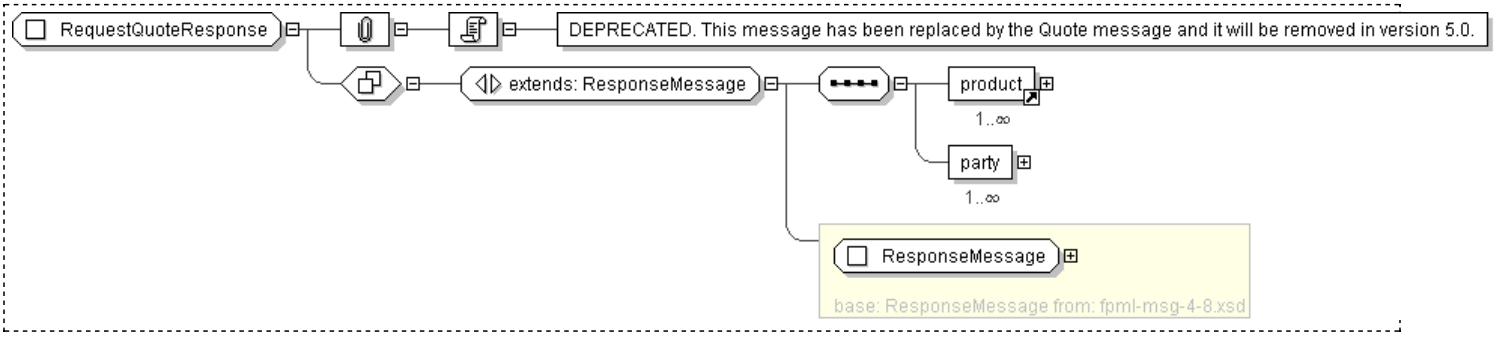
Super-types:	ResponseMessage < RequestQuoteResponse (by extension)
Sub-types:	None

Name	RequestQuoteResponse
Abstract	no
Documentation	DEPRECATED. This message has been replaced by the Quote message and it will be removed in version 5.0.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <product> ... </product> [1..*]
   <party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestQuoteResponse" deprecated="true" deprecatedReason="The naming of the message was not
clear, it has been replaced by the Quote message">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element ref=" product " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: AllegedCashflow
  - Complex Type: AssertedCashflow
  - Complex Type: AssertedPosition
  - Complex Type: CalculationDetails
  - Complex Type: CancelTradeCashflows
  - Complex Type: CashflowCalculationElements
  - Complex Type: CashflowCalculationPeriod
  - Complex Type: CashflowFixing
  - Complex Type: CashflowFixingReference
  - Complex Type: CashflowId
  - Complex Type: CashflowNotional
  - Complex Type: CashflowObservation
  - Complex Type: CashflowObservationReference
  - Complex Type: DefinePosition
  - Complex Type: GrossCashflow
  - Complex Type: InitialPortfolioDefinition
  - Complex Type: MatchId
  - Complex Type: PaymentDetails
  - Complex Type: PaymentId
  - Complex Type: PaymentMatching
  - Complex Type: PortfolioDefinition
  - Complex Type: PositionMatchResult
  - Complex Type: PositionMatchStatus
  - Complex Type: PositionProposedMatch
  - Complex Type: PositionReference
  - Complex Type: PositionsAcknowledged
  - Complex Type: PositionsAsserted
  - Complex Type: PositionsMatchResults
  - Complex Type: RequestPortfolio
  - Complex Type: StepReference
  - Complex Type: TradeCashflowsAsserted
  - Complex Type: TradeCashflowsId
  - Complex Type: TradeCashflowsMatchResult
  - Complex Type: TradeCashflowsProposedMatch
  - Complex Type: TradeCashflowsStatus
  - Complex Type: TradeDetails
  - Complex Type: TradeIdentifyingItems
  - Complex Type: TradeUnderlyer
  - Complex Type: TradeUnderlyerReference
  - Complex Type: UnderlyerReferenceUnits
  - Complex Type: UnprocessedPosition
  - Model Group: DefinitionAndCashflows.model
  - Model Group: IdAndTradeCashflows.model
  - Model Group: PositionWithoutId.model
  - Model Group: TradeCashflows.model
  - Model Group: TradeCashflowsDefinition.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-reporting-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-reporting-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:
- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: DefinitionAndCashflows.model

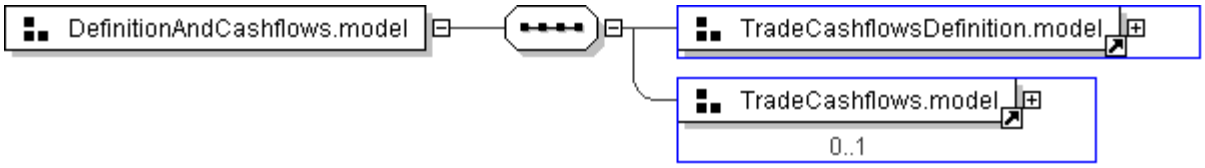
[Table of contents]

Name	DefinitionAndCashflows.model
Used by (from the same schema document)	Complex Type AllegedCashflow , Complex Type AssertedCashflow

XML Instance Representation

```
<asOfDate> xsd:dateTime </asOfDate> [0..1]
'The date and time at which the set of cashflows was defined.'

<tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
'Unique identifier assigned by the party asserting the set of cashflows to be
reconciled.'

Start Group: TradeCashflows.model [0..1]
   <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
   'Structure that holds reference to the trade through the tradeId and optionally
   some trade-specific elements for identifying the trade in the case of trades
   that have not been negotiated through electronic platforms and for which the
   counterparty\'s trade ID has not been captured.'

   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
   'The adjusted date in which the payments are being paid/received.'

   <payment> PaymentMatching </payment> [1..*]
   'Specifies the payment that is exposed to the matching process. Usually there
   will be a single payment but for cross-currency swaps a different payment per
   currency shall be provided.'

End Group: TradeCashflows.model
```

Diagram



Schema Component Representation

```
<xsd:group name="DefinitionAndCashflows.model">
   <xsd:sequence>
      <xsd:group ref=" TradeCashflowsDefinition.model "/>
      <xsd:group ref=" TradeCashflows.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: IdAndTradeCashflows.model

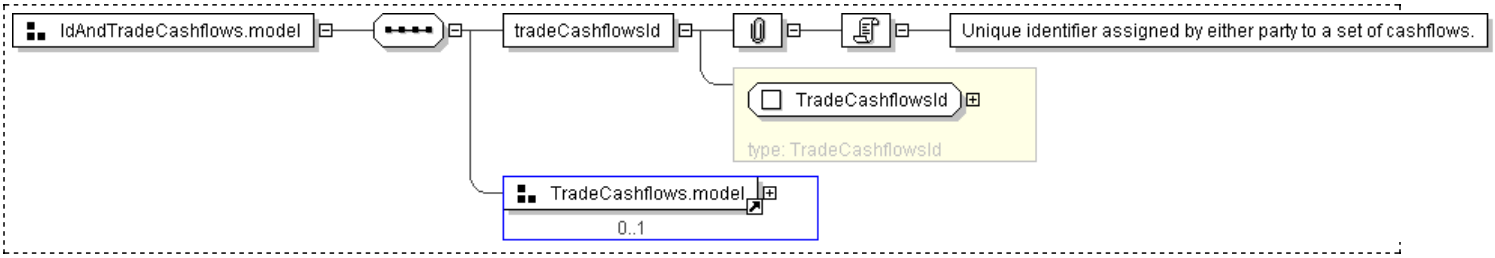
[Table of contents]

Name	IdAndTradeCashflows.model
Used by (from the same schema document)	Complex Type CancelTradeCashflows , Complex Type TradeCashflowsProposedMatch

XML Instance Representation

```
<tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
'Unique identifier assigned by either party to a set of cashflows.'

Start Group: TradeCashflows.model [0..1]
   <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
   'Structure that holds reference to the trade through the tradeId and optionally some trade-specific elements
   for identifying the trade in the case of trades that have not been negotiated through electronic platforms
   and for which the counterparty\'s trade ID has not been captured.'

   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
   'The adjusted date in which the payments are being paid/received.'

   <payment> PaymentMatching </payment> [1..*]
   'Specifies the payment that is exposed to the matching process. Usually there will be a single payment but
   for cross-currency swaps a different payment per currency shall be provided.'

End Group: TradeCashflows.model
```

Diagram



Schema Component Representation

```
<xsd:group name="IdAndTradeCashflows.model">
   <xsd:sequence>
      <xsd:element name="tradeCashflowsId" type=" TradeCashflowsId "/>
      <xsd:group ref=" TradeCashflows.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PositionWithoutId.model

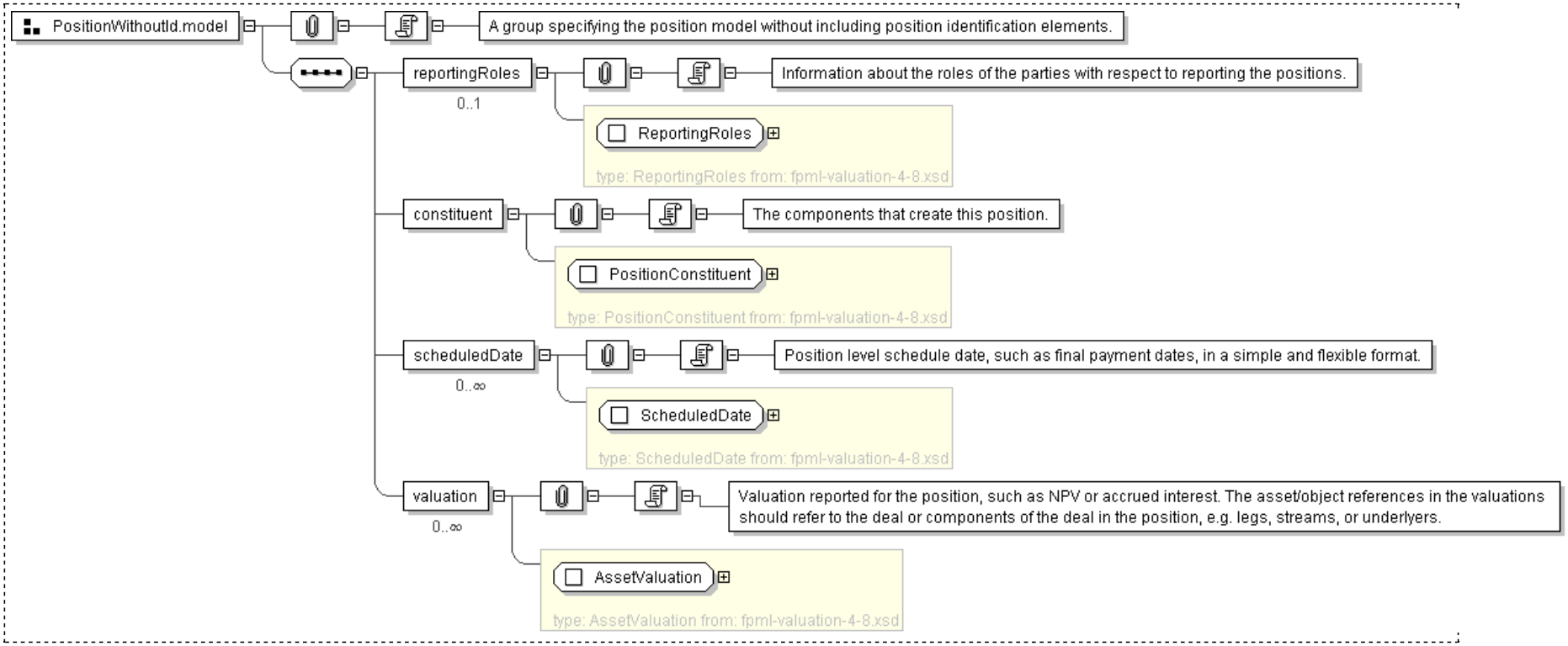
[Table of contents]

Name	PositionWithoutId.model
Used by (from the same schema document)	Complex Type AssertedPosition , Complex Type PositionProposedMatch
Documentation	A group specifying the position model without including position identification elements.

XML Instance Representation

```
<reportingRoles> ReportingRoles </reportingRoles> [0..1]
'Information about the roles of the parties with respect to reporting the positions.'

<constituent> PositionConstituent </constituent> [1]
'The components that create this position.'

<scheduledDate> ScheduledDate </scheduledDate> [0..*]
'Position level schedule date, such as final payment dates, in a simple and flexible format.'

<valuation> AssetValuation </valuation> [0..*]
'Valuation reported for the position, such as NPV or accrued interest. The asset/object references in the valuations should refer to
the deal or components of the deal in the position, e.g. legs, streams, or underlyers.'
```

Diagram



Schema Component Representation

```
<xsd:group name="PositionWithoutId.model">
   <xsd:sequence>
      <xsd:element name="reportingRoles" type=" ReportingRoles " minOccurs="0"/>
      <xsd:element name="constituent" type=" PositionConstituent "/>
      <xsd:element name="scheduledDate" type=" ScheduledDate " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="valuation" type=" AssetValuation " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: TradeCashflows.model

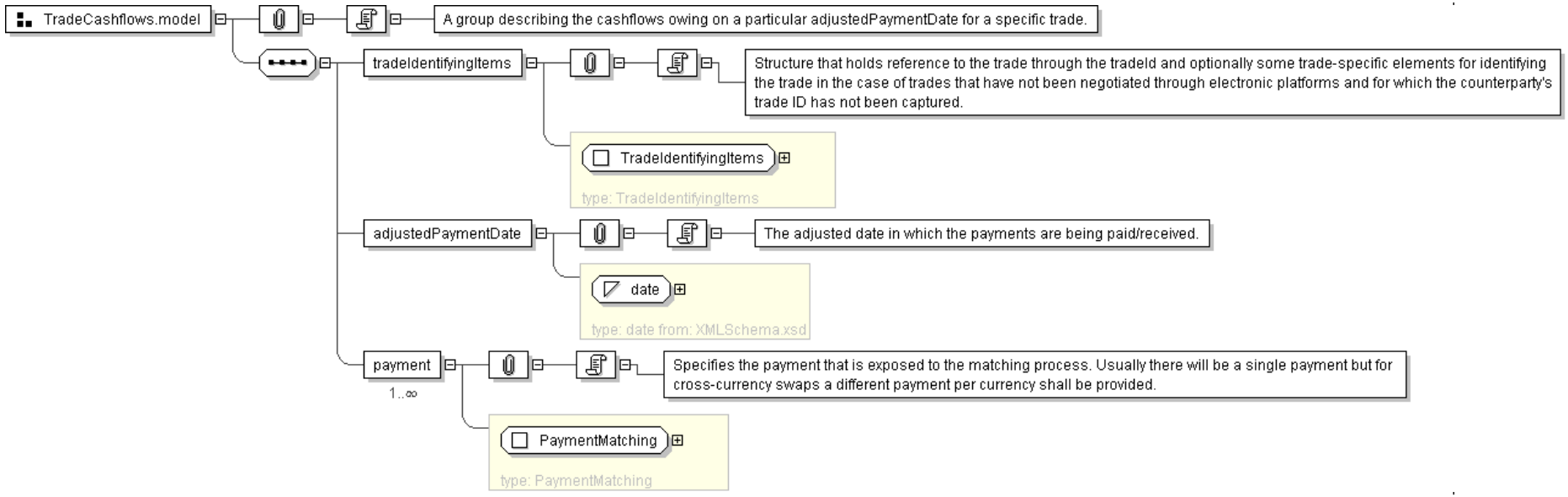
[Table of contents]

Name	TradeCashflows.model
Used by (from the same schema document)	Complex Type TradeCashflowsAsserted , Model Group DefinitionAndCashflows.model , Model Group IdAndTradeCashflows.model
Documentation	A group describing the cashflows owing on a particular adjustedPaymentDate for a specific trade.

XML Instance Representation

```
<tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
'Structure that holds reference to the trade through the tradeId and optionally some trade-specific elements for identifying the trade in the
case of trades that have not been negotiated through electronic platforms and for which the counterparty\'s trade ID has not been captured.'

<adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
'The adjusted date in which the payments are being paid/received.'

<payment> PaymentMatching </payment> [1..*]
'Specifies the payment that is exposed to the matching process. Usually there will be a single payment but for cross-currency swaps a different
payment per currency shall be provided.'
```

Diagram



Schema Component Representation

```
<xsd:group name="TradeCashflows.model">
  <xsd:sequence>
    <xsd:element name="tradeIdentifyingItems" type=" TradeIdentifyingItems "/>
    <xsd:element name="adjustedPaymentDate" type=" xsd:date "/>
    <xsd:element name="payment" type=" PaymentMatching " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: TradeCashflowsDefinition.model

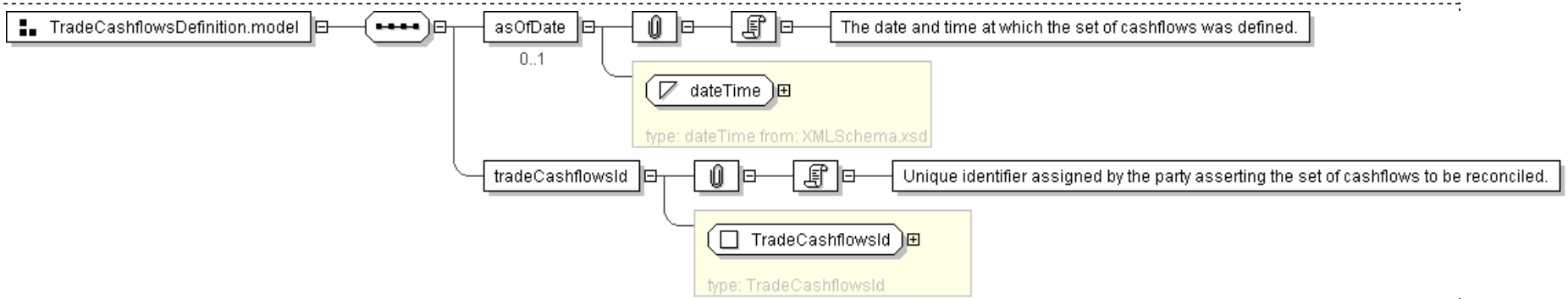
[Table of contents]

Name	TradeCashflowsDefinition.model
Used by (from the same schema document)	Complex Type TradeCashflowsAsserted , Model Group DefinitionAndCashflows.model

XML Instance Representation

```
<asOfDate> xsd:dateTime </asOfDate> [0..1]
'The date and time at which the set of cashflows was defined.'

<tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
'Unique identifier assigned by the party asserting the set of cashflows to be reconciled.'
```

Diagram



Schema Component Representation

```
<xsd:group name="TradeCashflowsDefinition.model">
  <xsd:sequence>
    <xsd:element name="asOfDate" type=" xsd:dateTime " minOccurs="0"/>
    <xsd:element name="tradeCashflowsId" type=" TradeCashflowsId "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AllegedCashflow

[Table of contents]

Super-types:	None
Sub-types:	None

Name	AllegedCashflow
Used by (from the same schema document)	Complex Type TradeCashflowsMatchResult
Abstract	no

XML Instance Representation

```
<...>
   <asOfDate> xsd:dateTime </asOfDate> [0..1]
    'The date and time at which the set of cashflows was defined.'

   <tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
    'Unique identifier assigned by the party asserting the set of cashflows to be
    reconciled.'

   Start Group: TradeCashflows.model [0..1]
      <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
       'Structure that holds reference to the trade through the tradeId and
       optionally some trade-specific elements for identifying the trade in the case
       of trades that have not been negotiated through electronic platforms and for
       which the counterparty\'s trade ID has not been captured.'

      <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
       'The adjusted date in which the payments are being paid/received.'

      <payment> PaymentMatching </payment> [1..*]
       'Specifies the payment that is exposed to the matching process. Usually there
       will be a single payment but for cross-currency swaps a different payment per
       currency shall be provided.'

   End Group: TradeCashflows.model
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="AllegedCashflow">
   <xsd:sequence>
      <xsd:group ref=" DefinitionAndCashflows.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssertedCashflow

[Table of contents]

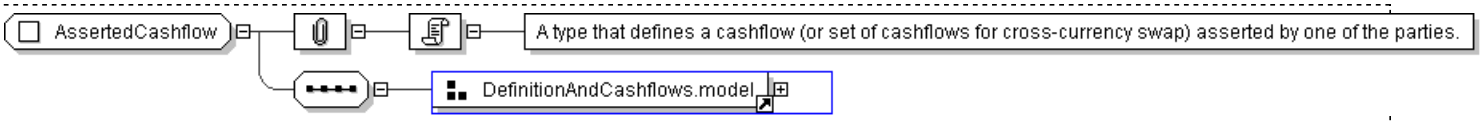
Super-types:	None
Sub-types:	None

Name	AssertedCashflow
Used by (from the same schema document)	Complex Type TradeCashflowsMatchResult
Abstract	no
Documentation	A type that defines a cashflow (or set of cashflows for cross-currency swap) asserted by one of the parties.

XML Instance Representation

```
<...>
   <asOfDate> xsd:dateTime </asOfDate> [0..1]
    'The date and time at which the set of cashflows was defined.'

   <tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
    'Unique identifier assigned by the party asserting the set of cashflows to be reconciled.'

   Start Group: TradeCashflows.model [0..1]
      <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
       'Structure that holds reference to the trade through the tradeId and optionally some trade-specific
       elements for identifying the trade in the case of trades that have not been negotiated through
       electronic platforms and for which the counterparty\'s trade ID has not been captured.'

      <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
       'The adjusted date in which the payments are being paid/received.'

      <payment> PaymentMatching </payment> [1..*]
       'Specifies the payment that is exposed to the matching process. Usually there will be a single
       payment but for cross-currency swaps a different payment per currency shall be provided.'

   End Group: TradeCashflows.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssertedCashflow">
   <xsd:sequence>
      <xsd:group ref=" DefinitionAndCashflows.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssertedPosition

[Table of contents]

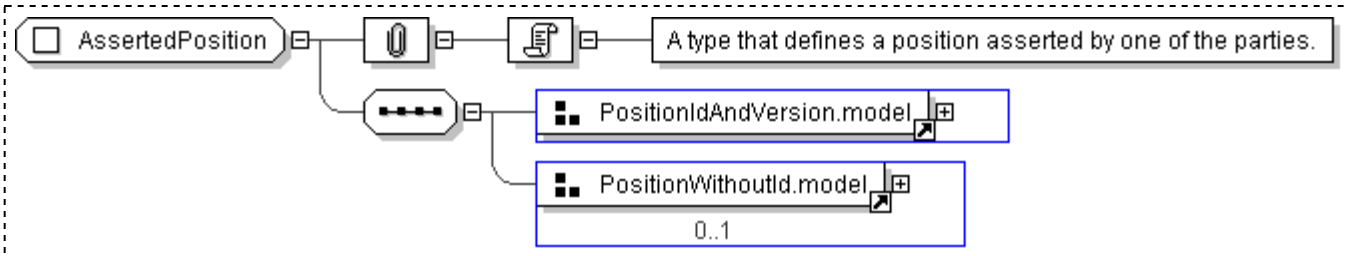
Super-types:	None
Sub-types:	None

Name	AssertedPosition
Used by (from the same schema document)	Complex Type PositionMatchResult , Complex Type PositionMatchResult
Abstract	no
Documentation	A type that defines a position asserted by one of the parties.

XML Instance Representation

```
<...>
   <positionId> PositionId </positionId> [1]
   'A version-independent identifier for the position, possibly based on trade
   identifier.'

   <version> xsd:positiveInteger </version> [0..1]
   'A version identifier. Version identifiers must be ascending, i.e. higher
   numbers imply newer versions. There is no requirement that version identifiers
   for a position be sequential or small, so for example timestamp-based version
   identifiers could be used.'

   Start Group: PositionWithoutId.model [0..1]
      <reportingRoles> ReportingRoles </reportingRoles> [0..1]
      'Information about the roles of the parties with respect to reporting the
      positions.'

      <constituent> PositionConstituent </constituent> [1]
      'The components that create this position.'

      <scheduledDate> ScheduledDate </scheduledDate> [0..*]
      'Position level schedule date, such as final payment dates, in a simple and
      flexible format.'

      <valuation> AssetValuation </valuation> [0..*]
      'Valuation reported for the position, such as NPV or accrued interest. The
      asset/object references in the valuations should refer to the deal or
      components of the deal in the position, e.g. legs, streams, or underlyers.'

   End Group: PositionWithoutId.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssertedPosition">
```

```
   <xsd:sequence>
      <xsd:group ref=" PositionIdAndVersion.model "/>
      <xsd:group ref=" PositionWithoutId.model " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationDetails

[Table of contents]

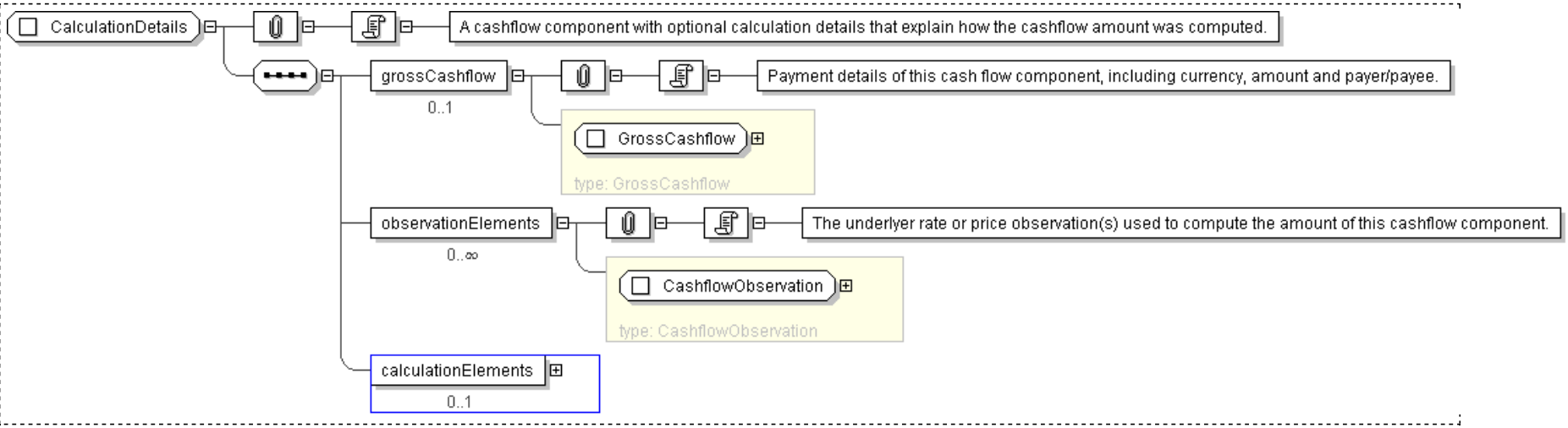
Super-types:	None
Sub-types:	None

Name	CalculationDetails
Used by (from the same schema document)	Complex Type PaymentMatching
Abstract	no
Documentation	A cashflow component with optional calculation details that explain how the cashflow amount was computed.

XML Instance Representation

```
<...>
  <grossCashflow> GrossCashflow </grossCashflow> [0..1]
  'Payment details of this cash flow component, including currency, amount and payer/payee.'

  <observationElements> CashflowObservation </observationElements> [0..*]
  'The underlyer rate or price observation(s) used to compute the amount of this cashflow component.'

  <calculationElements> CashflowCalculationElements </calculationElements> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationDetails">
  <xsd:sequence>
    <xsd:element name="grossCashflow" type=" GrossCashflow " minOccurs="0"/>
    <xsd:element name="observationElements" type=" CashflowObservation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="calculationElements" type=" CashflowCalculationElements " minOccurs="0"/>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancelTradeCashflows

[Table of contents]

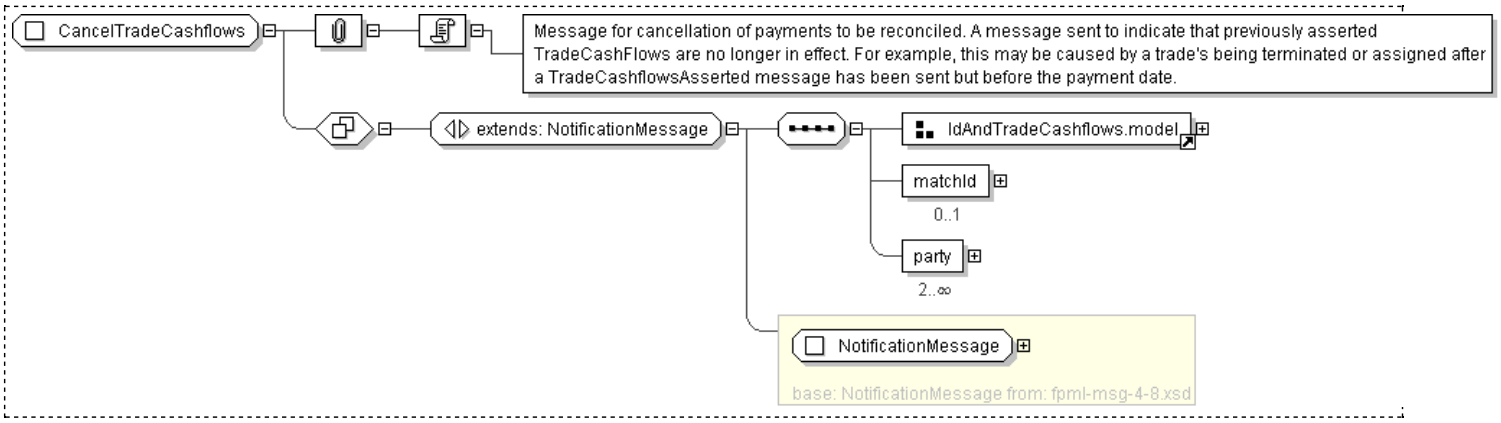
Super-types:	NotificationMessage < CancelTradeCashflows (by extension)
Sub-types:	None

Name	CancelTradeCashflows
Abstract	no
Documentation	Message for cancellation of payments to be reconciled. A message sent to indicate that previously asserted TradeCashFlows are no longer in effect. For example, this may be caused by a trade's being terminated or assigned after a TradeCashflowsAsserted message has been sent but before the payment date.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead,
 it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build
 number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a
 version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases
 within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
   'Unique identifier assigned by either party to a set of cashflows.'

   Start Group: TradeCashflows.model [0..1]
      <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
      'Structure that holds reference to the trade through the tradeId and optionally some trade-specific elements
      for identifying the trade in the case of trades that have not been negotiated through electronic platforms and
      for which the counterparty\'s trade ID has not been captured.'

      <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
      'The adjusted date in which the payments are being paid/received.'

      <payment> PaymentMatching </payment> [1..*]
      'Specifies the payment that is exposed to the matching process. Usually there will be a single payment but for
      cross-currency swaps a different payment per currency shall be provided.'

   End Group: TradeCashflows.model
   <matchId> MatchId </matchId> [0..1]
   'A unique identifier assigned by matching service to each set of matched cashflows.'

   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancelTradeCashflows">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
```

```
        <xsd:sequence>
          <xsd:group ref=" IdAndTradeCashflows.model "/>
          <xsd:element name="matchId" type=" MatchId " minOccurs="0"/>
          <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowCalculationElements

[Table of contents]

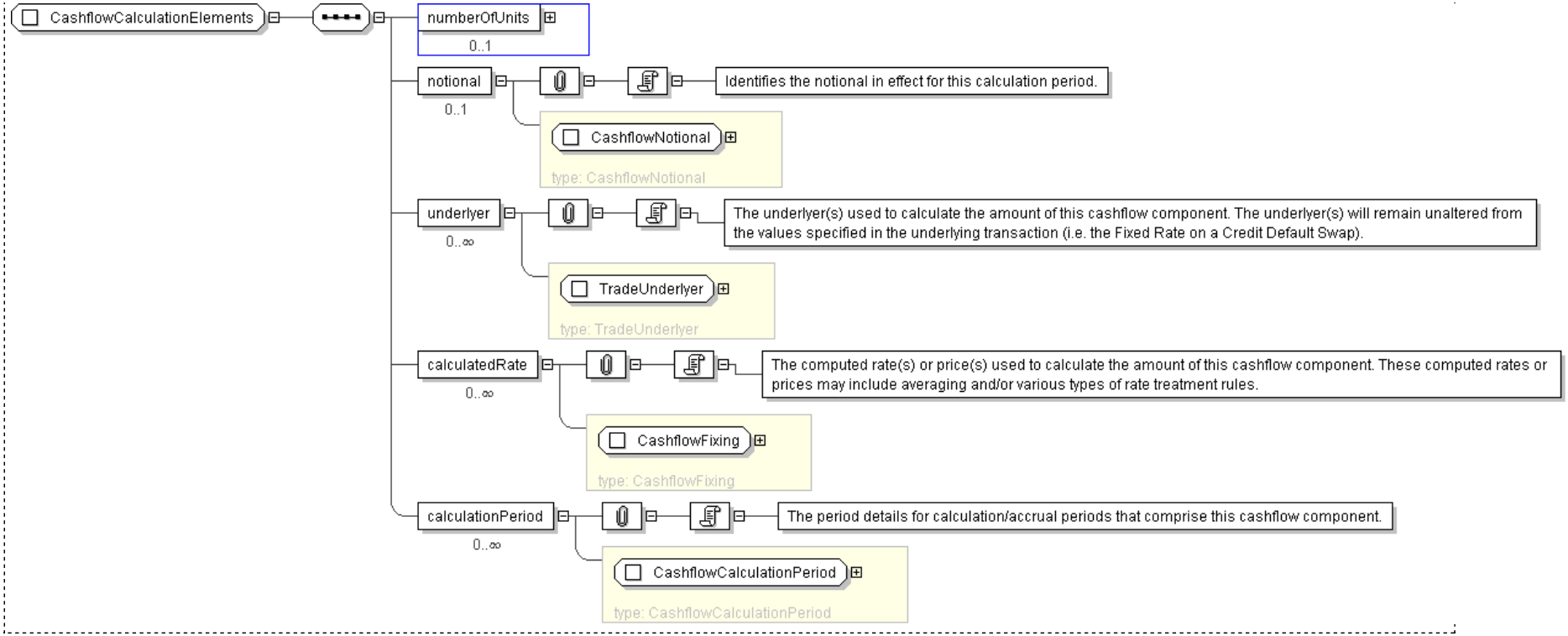
Super-types:	None
Sub-types:	None

Name	CashflowCalculationElements
Used by (from the same schema document)	Complex Type CalculationDetails
Abstract	no

XML Instance Representation

```
<...>
   <numberOfUnits> UnderlyerReferenceUnits </numberOfUnits> [0..1]
   <notional> CashflowNotional </notional> [0..1]
   'Identifies the notional in effect for this calculation period.'

   <underlyer> TradeUnderlyer </underlyer> [0..*]
   'The underlyer(s) used to calculate the amount of this cashflow component. The underlyer(s) will remain unaltered from the values specified in
   the underlying transaction (i.e. the Fixed Rate on a Credit Default Swap).'

   <calculatedRate> CashflowFixing </calculatedRate> [0..*]
   'The computed rate(s) or price(s) used to calculate the amount of this cashflow component. These computed rates or prices may include
   averaging and/or various types of rate treatment rules.'

   <calculationPeriod> CashflowCalculationPeriod </calculationPeriod> [0..*]
   'The period details for calculation/accrual periods that comprise this cashflow component.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowCalculationElements">
  <xsd:sequence>
    <xsd:element name="numberOfUnits" type=" UnderlyerReferenceUnits " minOccurs="0"/>
    <xsd:element name="notional" type=" CashflowNotional " minOccurs="0"/>
    <xsd:element name="underlyer" type=" TradeUnderlyer " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="calculatedRate" type=" CashflowFixing " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="calculationPeriod" type=" CashflowCalculationPeriod " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowCalculationPeriod

[Table of contents]

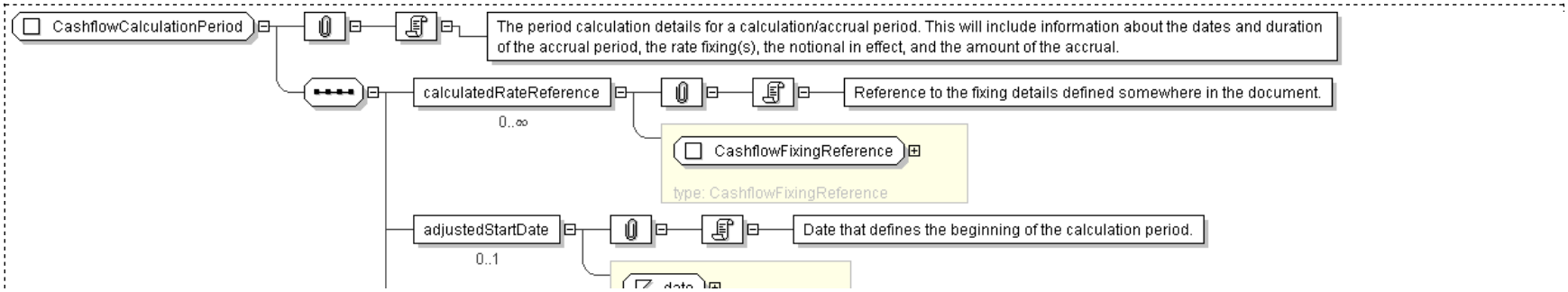
Super-types:	None
Sub-types:	None

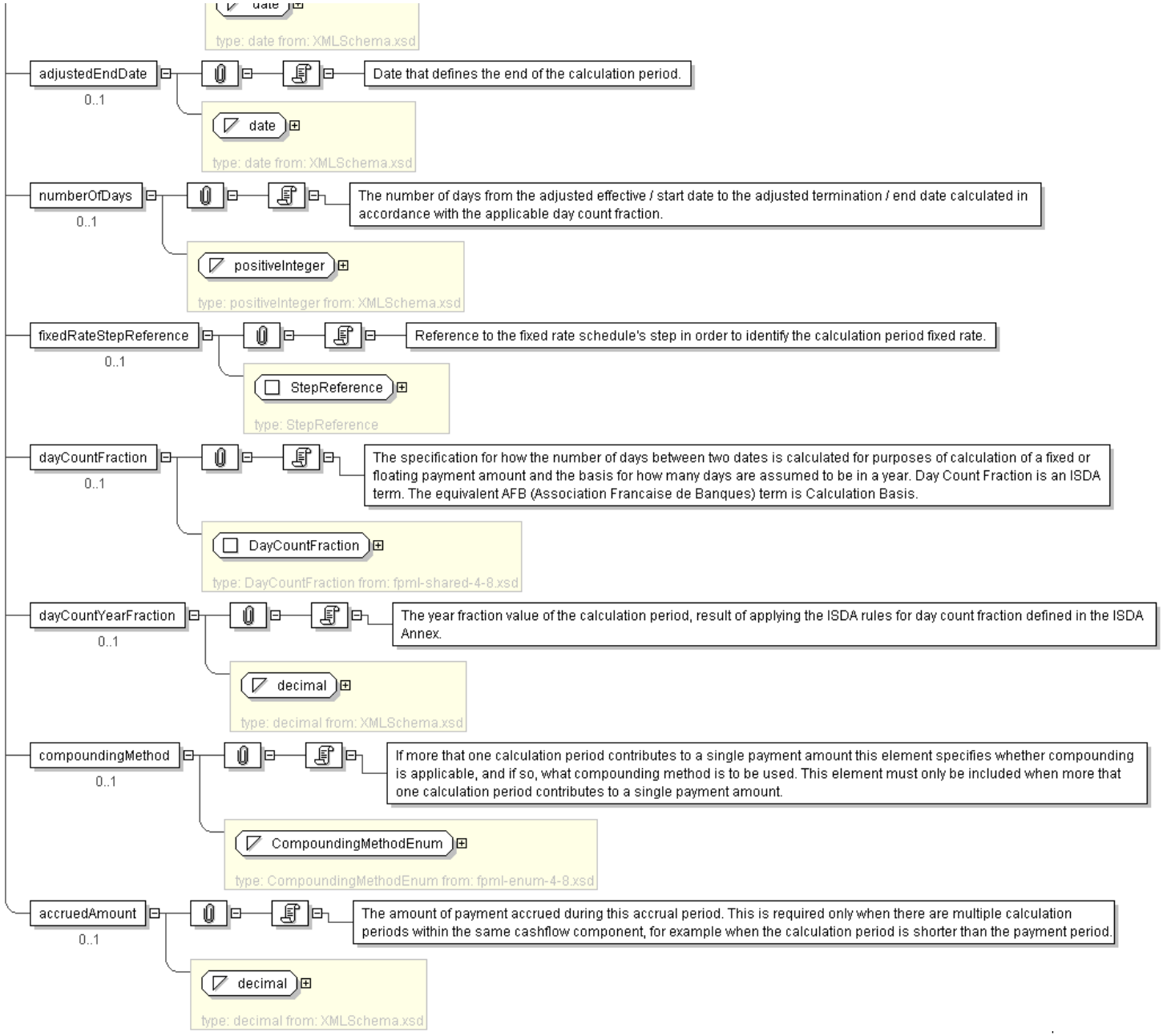
Name	CashflowCalculationPeriod
Used by (from the same schema document)	Complex Type CashflowCalculationElements
Abstract	no
Documentation	The period calculation details for a calculation/accrual period. This will include information about the dates and duration of the accrual period, the rate fixing(s), the notional in effect, and the amount of the accrual.

XML Instance Representation

```
<...>
   <calculatedRateReference> CashflowFixingReference </calculatedRateReference> [0..*]
   'Reference to the fixing details defined somewhere in the document.'

   <adjustedStartDate> xsd:date </adjustedStartDate> [0..1]
   'Date that defines the beginning of the calculation period.'

   <adjustedEndDate> xsd:date </adjustedEndDate> [0..1]
   'Date that defines the end of the calculation period.'

   <numberOfDays> xsd:positiveInteger </numberOfDays> [0..1]
   'The number of days from the adjusted effective / start date to the adjusted termination / end date calculated in accordance with the applicable
   day count fraction.'

   <fixedRateStepReference> StepReference </fixedRateStepReference> [0..1]
   'Reference to the fixed rate schedule\'s step in order to identify the calculation period fixed rate.'

   <dayCountFraction> DayCountFraction </dayCountFraction> [0..1]
   'The specification for how the number of days between two dates is calculated for purposes of calculation of a fixed or floating payment amount and
   the basis for how many days are assumed to be in a year. Day Count Fraction is an ISDA term. The equivalent AFB (Association Francaise de Banques)
   term is Calculation Basis.'

   <dayCountYearFraction> xsd:decimal </dayCountYearFraction> [0..1]
   'The year fraction value of the calculation period, result of applying the ISDA rules for day count fraction defined in the ISDA Annex.'

   <compoundingMethod> CompoundingMethodEnum </compoundingMethod> [0..1]
   'If more that one calculation period contributes to a single payment amount this element specifies whether compounding is applicable, and if so,
   what compounding method is to be used. This element must only be included when more that one calculation period contributes to a single payment
   amount.'

   <accruedAmount> xsd:decimal </accruedAmount> [0..1]
   'The amount of payment accrued during this accrual period. This is required only when there are multiple calculation periods within the same
   cashflow component, for example when the calculation period is shorter than the payment period.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="CashflowCalculationPeriod">
  <xsd:sequence>
    <xsd:element name="calculatedRateReference" type=" CashflowFixingReference " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="adjustedStartDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="adjustedEndDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="numberOfDays" type=" xsd:positiveInteger " minOccurs="0"/>
    <xsd:element name="fixedRateStepReference" type=" StepReference " minOccurs="0"/>
    <xsd:element name="dayCountFraction" type=" DayCountFraction " minOccurs="0"/>
    <xsd:element name="dayCountYearFraction" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="compoundingMethod" type=" CompoundingMethodEnum " minOccurs="0"/>
```

```
    <xsd:element name="accruedAmount" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowFixing

[Table of contents]

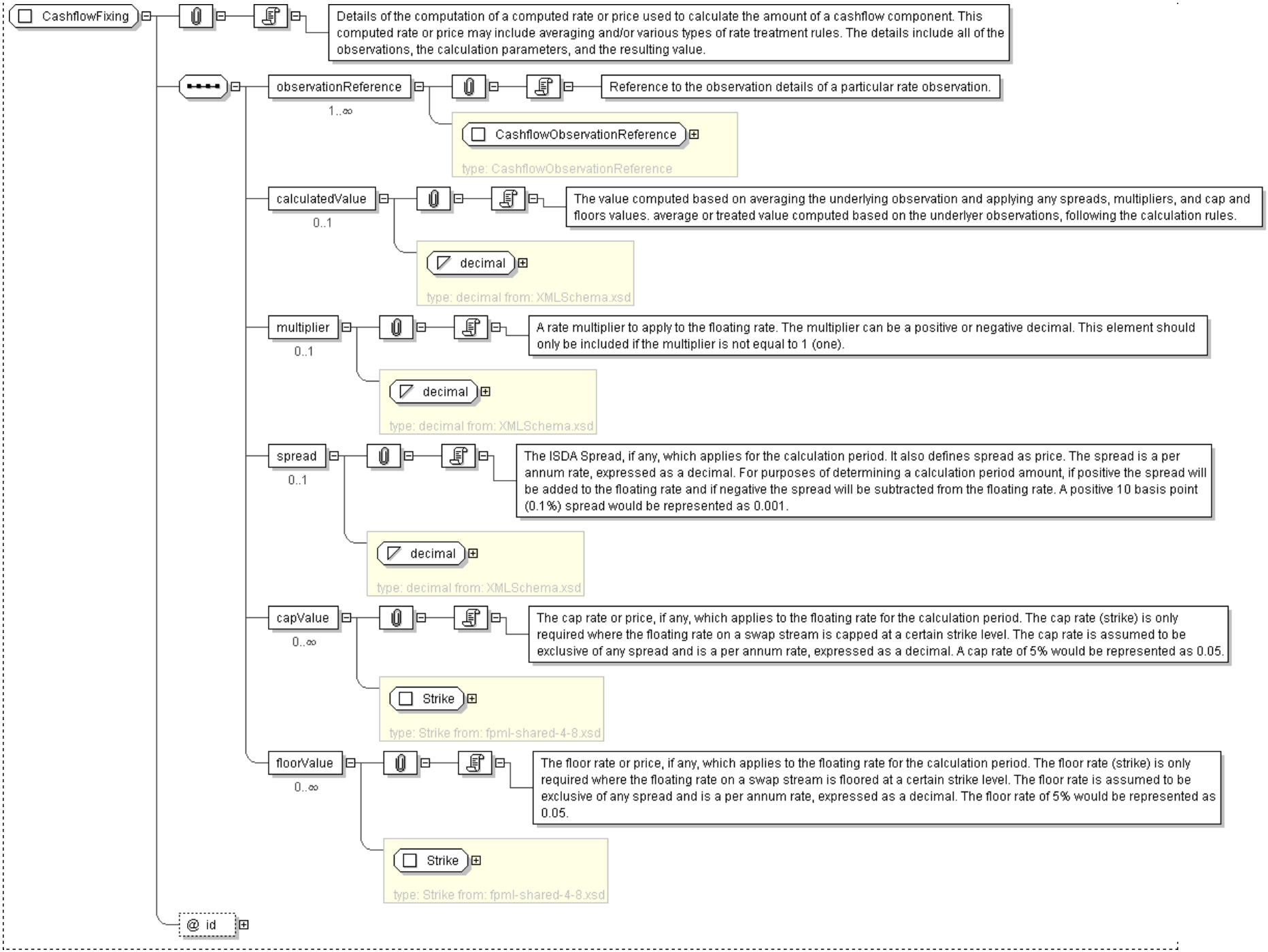
Super-types:	None
Sub-types:	None

Name	CashflowFixing
Used by (from the same schema document)	Complex Type CashflowCalculationElements
Abstract	no
Documentation	Details of the computation of a computed rate or price used to calculate the amount of a cashflow component. This computed rate or price may include averaging and/or various types of rate treatment rules. The details include all of the observations, the calculation parameters, and the resulting value.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <observationReference> CashflowObservationReference </observationReference> [1..*]
   'Reference to the observation details of a particular rate observation.'

   <calculatedValue> xsd:decimal </calculatedValue> [0..1]
   'The value computed based on averaging the underlying observation and applying any spreads, multipliers, and cap and floors values.
   average or treated value computed based on the underlyer observations, following the calculation rules.'

   <multiplier> xsd:decimal </multiplier> [0..1]
   'A rate multiplier to apply to the floating rate. The multiplier can be a positive or negative decimal. This element should only be
   included if the multiplier is not equal to 1 (one).'

   <spread> xsd:decimal </spread> [0..1]
   'The ISDA Spread, if any, which applies for the calculation period. It also defines spread as price. The spread is a per annum rate,
   expressed as a decimal. For purposes of determining a calculation period amount, if positive the spread will be added to the floating
   rate and if negative the spread will be subtracted from the floating rate. A positive 10 basis point (0.1%) spread would be
   represented as 0.001.'

   <capValue> Strike </capValue> [0..*]
   'The cap rate or price, if any, which applies to the floating rate for the calculation period. The cap rate (strike) is only required
   where the floating rate on a swap stream is capped at a certain strike level. The cap rate is assumed to be exclusive of any spread
   and is a per annum rate, expressed as a decimal. A cap rate of 5% would be represented as 0.05.'

   <floorValue> Strike </floorValue> [0..*]
   'The floor rate or price, if any, which applies to the floating rate for the calculation period. The floor rate (strike) is only
   required where the floating rate on a swap stream is floored at a certain strike level. The floor rate is assumed to be exclusive of
   any spread and is a per annum rate, expressed as a decimal. The floor rate of 5% would be represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowFixing">
  <xsd:sequence>
    <xsd:element name="observationReference" type=" CashflowObservationReference " maxOccurs="unbounded"/>
    <xsd:element name="calculatedValue" type=" xsd:decimal " minOccurs="0"/>
```

```
      <xsd:element name="multiplier" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="spread" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="capValue" type=" Strike " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="floorValue" type=" Strike " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowFixingReference

[Table of contents]

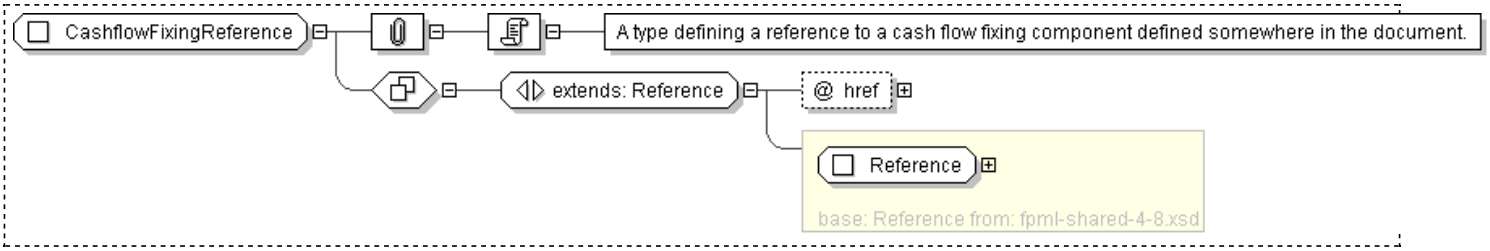
Super-types:	Reference < CashflowFixingReference (by extension)
Sub-types:	None

Name	CashflowFixingReference
Used by (from the same schema document)	Complex Type CashflowCalculationPeriod
Abstract	no
Documentation	A type defining a reference to a cash flow fixing component defined somewhere in the document.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowFixingReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="CashflowFixing"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowId

[Table of contents]

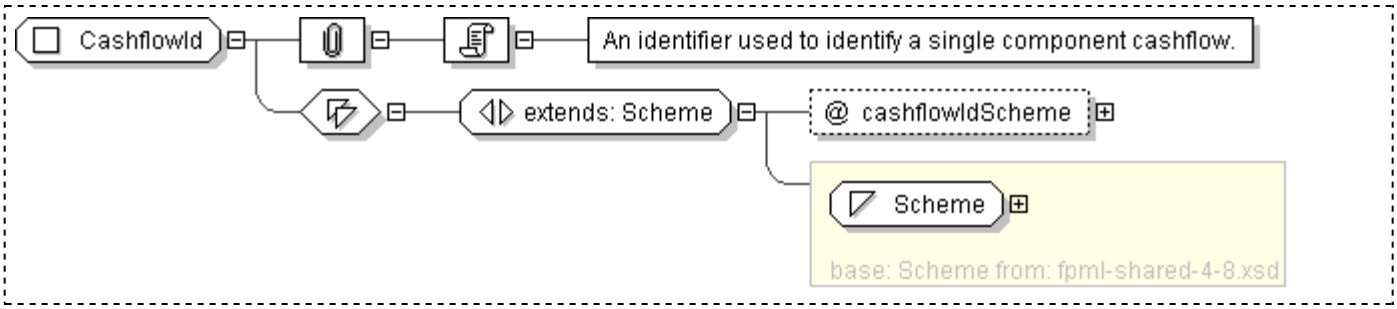
Super-types:	Scheme < CashflowId (by extension)
Sub-types:	None

Name	CashflowId
Used by (from the same schema document)	Complex Type GrossCashflow
Abstract	no
Documentation	An identifier used to identify a single component cashflow.

XML Instance Representation

```
<...
 cashflowIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="cashflowIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowNotional

[Table of contents]

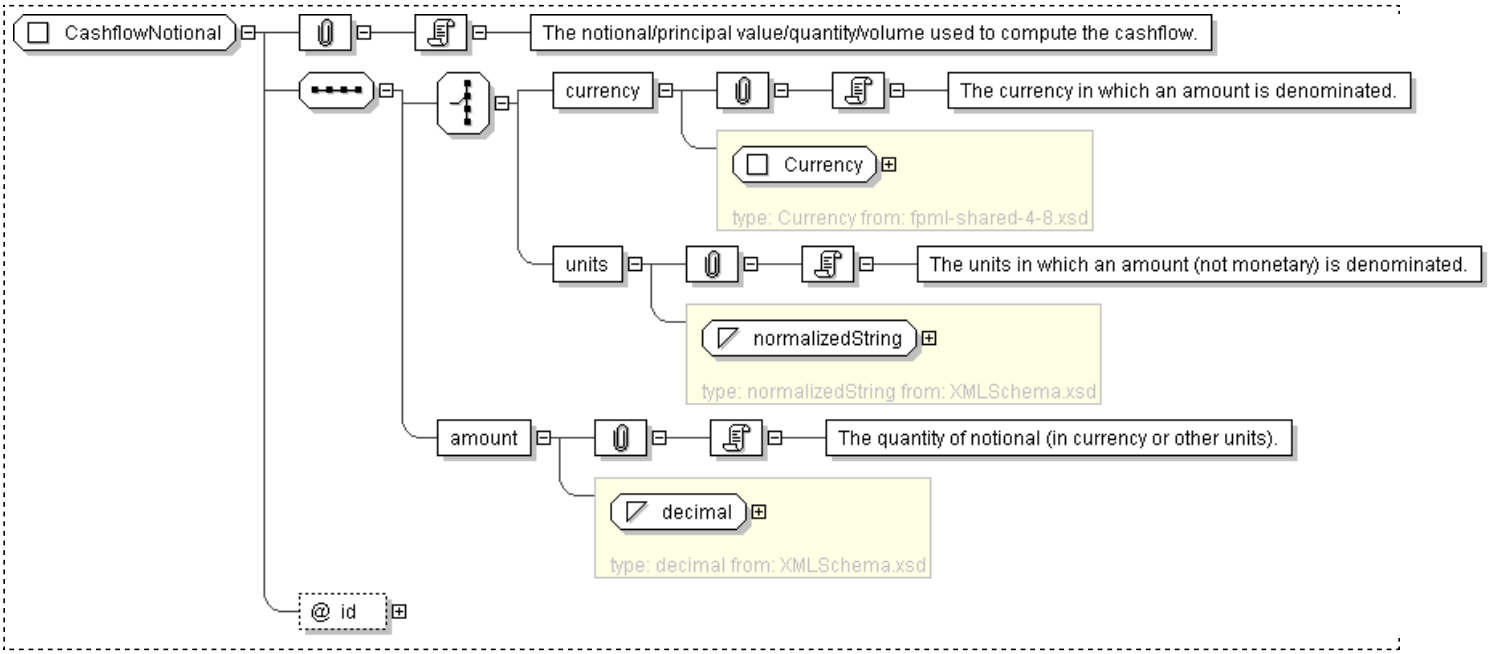
Super-types:	None
Sub-types:	None

Name	CashflowNotional
Used by (from the same schema document)	Complex Type CashflowCalculationElements , Complex Type TradeDetails
Abstract	no
Documentation	The notional/principal value/quantity/volume used to compute the cashflow.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <currency> Currency </currency> [1]
       'The currency in which an amount is denominated.'

      <units> xsd:normalizedString </units> [1]
       'The units in which an amount (not monetary) is denominated.'

   End Choice
   <amount> xsd:decimal </amount> [1]
   'The quantity of notional (in currency or other units).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowNotional">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="currency" type=" Currency "/>
         <xsd:element name="units" type=" xsd:normalizedString "/>
      </xsd:choice>
      <xsd:element name="amount" type=" xsd:decimal "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowObservation

[Table of contents]

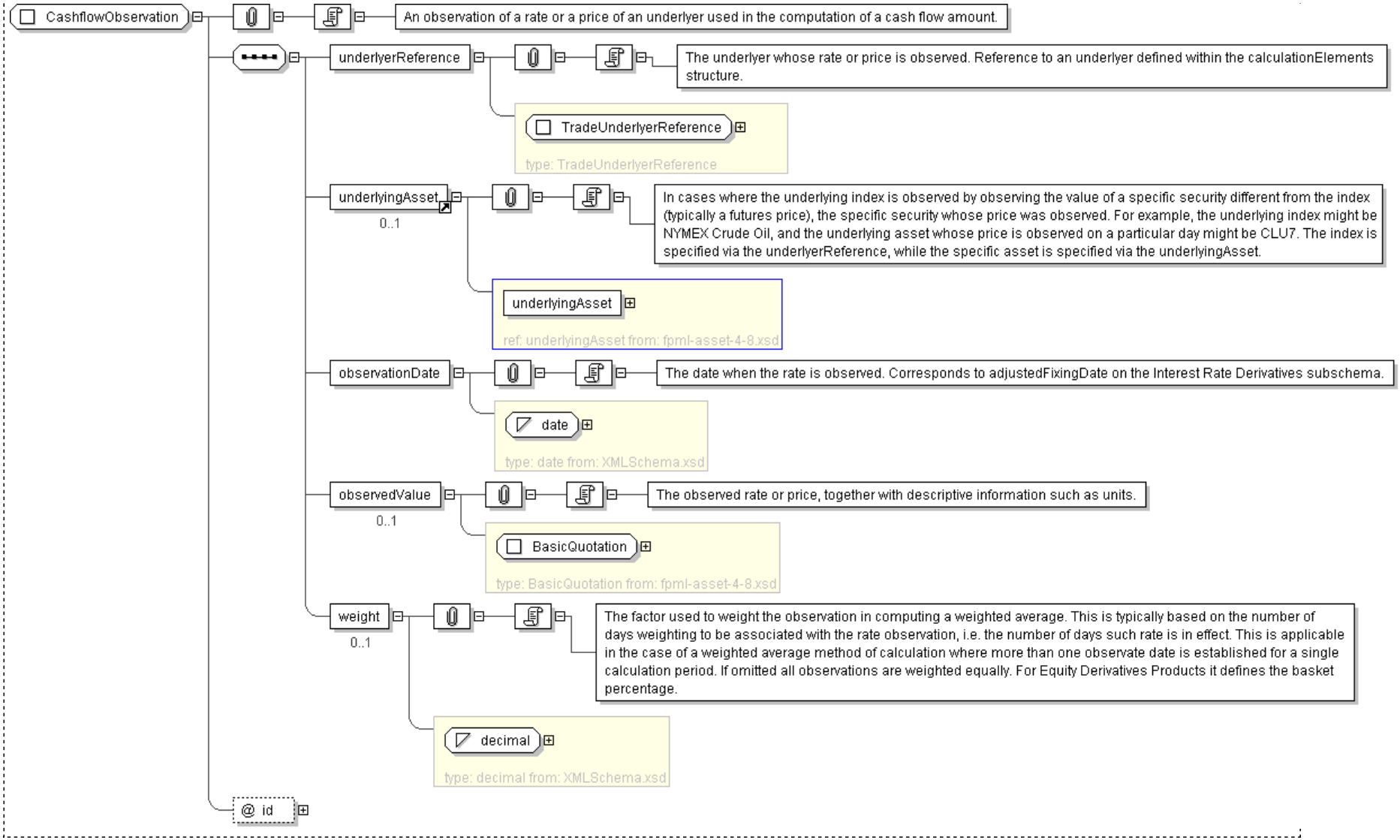
Super-types:	None
Sub-types:	None

Name	CashflowObservation
Used by (from the same schema document)	Complex Type CalculationDetails
Abstract	no
Documentation	An observation of a rate or a price of an underlyer used in the computation of a cash flow amount.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <underlyerReference> TradeUnderlyerReference </underlyerReference> [1]
   'The underlyer whose rate or price is observed. Reference to an underlyer defined within the calculationElements structure.'

   <underlyingAsset> ... </underlyingAsset> [0..1]
   'In cases where the underlying index is observed by observing the value of a specific security different from the index (typically a
   futures price), the specific security whose price was observed. For example, the underlying index might be NYMEX Crude Oil, and the
   underlying asset whose price is observed on a particular day might be CLU7. The index is specified via the underlyerReference, while the
   specific asset is specified via the underlyingAsset.'

   <observationDate> xsd:date </observationDate> [1]
   'The date when the rate is observed. Corresponds to adjustedFixingDate on the Interest Rate Derivatives subschema.'

   <observedValue> BasicQuotation </observedValue> [0..1]
   'The observed rate or price, together with descriptive information such as units.'

   <weight> xsd:decimal </weight> [0..1]
   'The factor used to weight the observation in computing a weighted average. This is typically based on the number of days weighting to be
   associated with the rate observation, i.e. the number of days such rate is in effect. This is applicable in the case of a weighted
   average method of calculation where more than one observate date is established for a single calculation period. If omitted all
   observations are weighted equally. For Equity Derivatives Products it defines the basket percentage.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowObservation">
  <xsd:sequence>
    <xsd:element name="underlyerReference" type=" TradeUnderlyerReference "/>
    <xsd:element ref=" underlyingAsset " minOccurs="0"/>
    <xsd:element name="observationDate" type=" xsd:date "/>
    <xsd:element name="observedValue" type=" BasicQuotation " minOccurs="0"/>
    <xsd:element name="weight" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowObservationReference

[Table of contents]

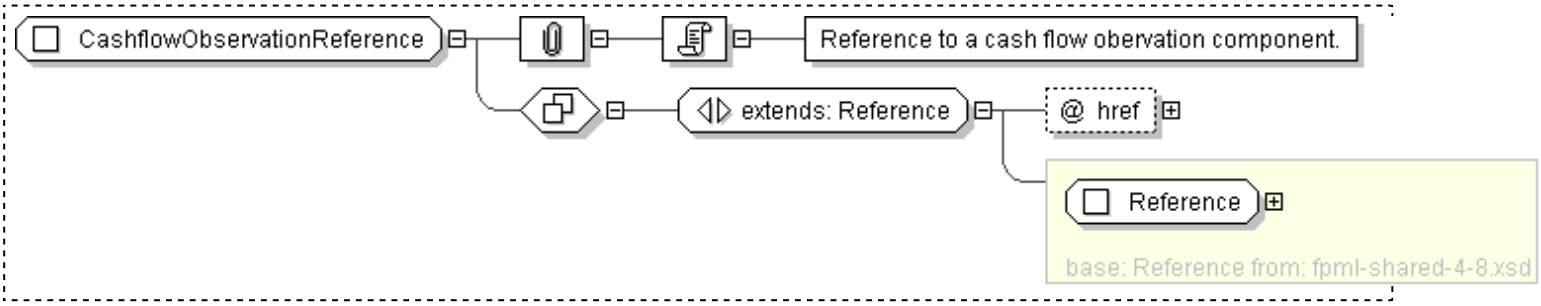
Super-types:	Reference < CashflowObservationReference (by extension)
Sub-types:	None

Name	CashflowObservationReference
Used by (from the same schema document)	Complex Type CashflowFixing
Abstract	no
Documentation	Reference to a cash flow obervation component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowObservationReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="CashflowObservation"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DefinePosition

[Table of contents]

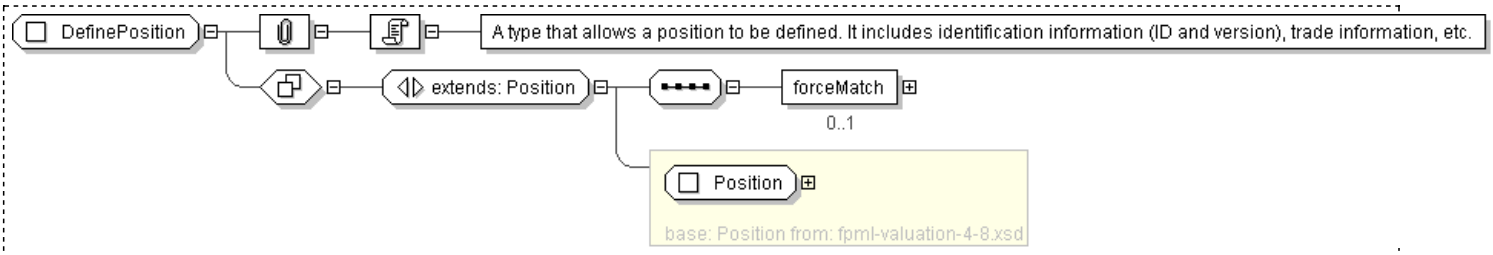
Super-types:	Position < DefinePosition (by extension)
Sub-types:	None

Name	DefinePosition
Used by (from the same schema document)	Complex Type PositionsAsserted , Complex Type PositionsAsserted
Abstract	no
Documentation	A type that allows a position to be defined. It includes identification information (ID and version), trade information, etc.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <positionId> PositionId </positionId> [1]
   'A version-independent identifier for the position, possibly based on trade identifier.'

   <version> xsd:positiveInteger </version> [0..1]
   'A version identifier. Version identifiers must be ascending, i.e. higher numbers imply newer versions.
   There is no requirement that version identifiers for a position be sequential or small, so for example
   timestamp-based version identifiers could be used.'

   <reportingRoles> ReportingRoles </reportingRoles> [0..1]
   'Information about the roles of the parties with respect to reporting the positions.'

   <constituent> PositionConstituent </constituent> [1]
   'The components that create this position.'

   <scheduledDate> ScheduledDate </scheduledDate> [0..*]
   'Position level schedule date, such as final payment dates, in a simple and flexible format.'

   <valuation> AssetValuation </valuation> [0..*]
   'Valuation reported for the position, such as NPV or accrued interest. The asset/object references in the
   valuations should refer to the deal or components of the deal in the position, e.g. legs, streams, or
   underlyers.'

   <forceMatch> PositionReference </forceMatch> [0..1]
   'An optional reference to a position supplied by the matching party that is known to match this one.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DefinePosition">
  <xsd:complexContent>
    <xsd:extension base=" Position ">
      <xsd:sequence>
        <xsd:element name="forceMatch" type=" PositionReference " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GrossCashflow

[Table of contents]

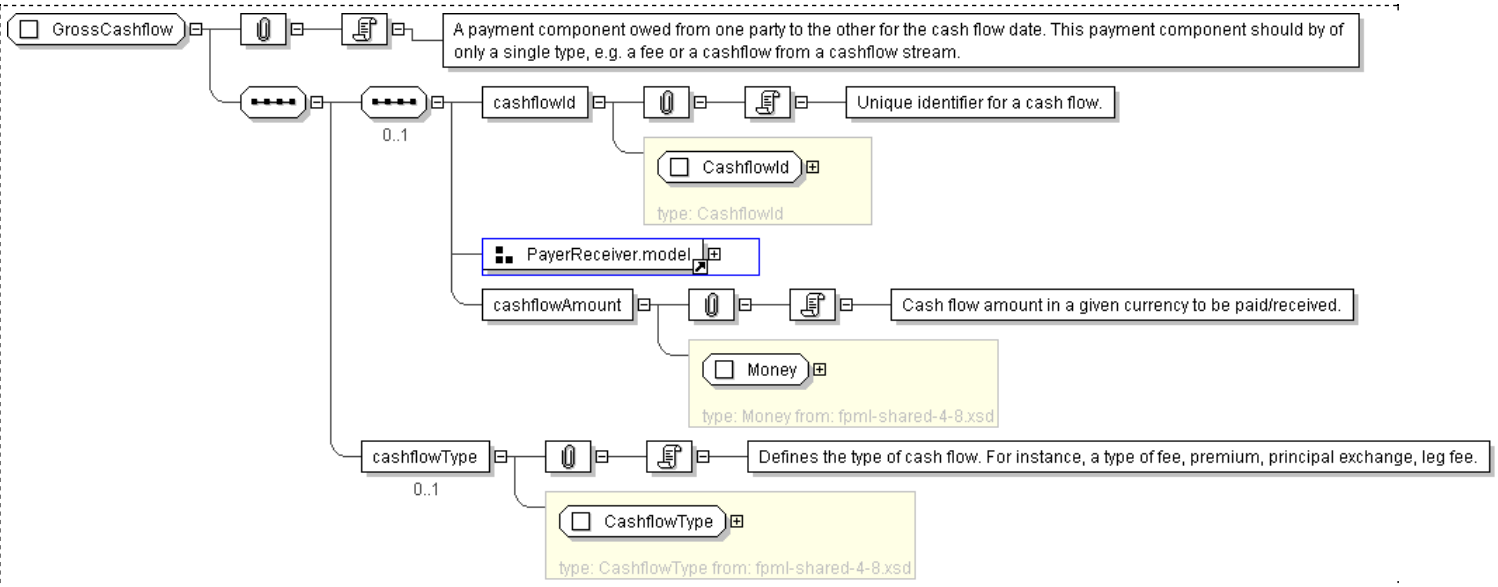
Super-types:	None
Sub-types:	None

Name	GrossCashflow
Used by (from the same schema document)	Complex Type CalculationDetails , Complex Type PaymentDetails
Abstract	no
Documentation	A payment component owed from one party to the other for the cash flow date. This payment component should by of only a single type, e.g. a fee or a cashflow from a cashflow stream.

XML Instance Representation

```
<...>
  Start Sequence [0..1]
    <cashflowId> CashflowId </cashflowId> [1]
    'Unique identifier for a cash flow.'

    <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

    <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

    <cashflowAmount> Money </cashflowAmount> [1]
    'Cash flow amount in a given currency to be paid/received.'

  End Sequence
  <cashflowType> CashflowType </cashflowType> [0..1]
  'Defines the type of cash flow. For instance, a type of fee, premium, principal exchange, leg fee.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GrossCashflow">
  <xsd:sequence>
    <xsd:sequence minOccurs="0">
      <xsd:element name="cashflowId" type=" CashflowId "/>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="cashflowAmount" type=" Money "/>
    </xsd:sequence>
    <xsd:element name="cashflowType" type=" CashflowType " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InitialPortfolioDefinition

[Table of contents]

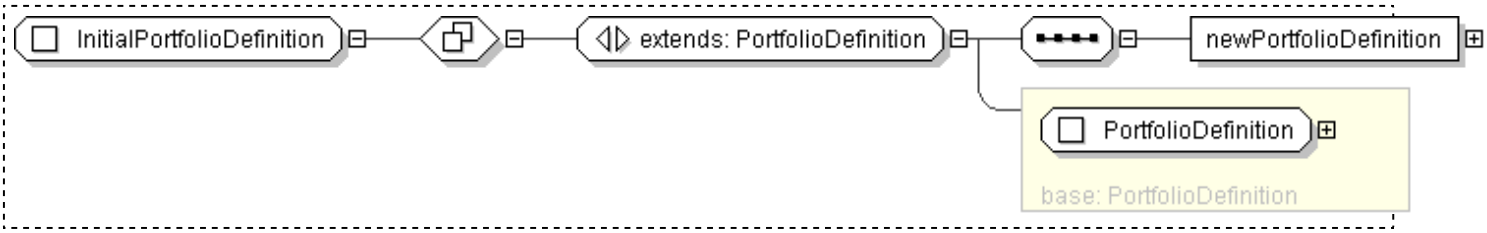
Super-types:	PortfolioDefinition < InitialPortfolioDefinition (by extension)
Sub-types:	None

Name	InitialPortfolioDefinition
Used by (from the same schema document)	Complex Type PositionsAsserted
Abstract	no

XML Instance Representation

```
<...>
   <portfolioName> xsd:normalizedString </portfolioName> [1]
   <asOfDate> xsd:date </asOfDate> [1]
   <definingParty> PartyReference </definingParty> [1]
   <matchingParty> PartyReference </matchingParty> [0..1]
   <newPortfolioDefinition> xsd:boolean </newPortfolioDefinition> [1]
   'Indicates whether it\'s a definition of a new portfolio (true) or an update to
   an existing one (false).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InitialPortfolioDefinition">
   <xsd:complexContent>
      <xsd:extension base=" PortfolioDefinition ">
         <xsd:sequence>
            <xsd:element name="newPortfolioDefinition" type=" xsd:boolean "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MatchId

[Table of contents]

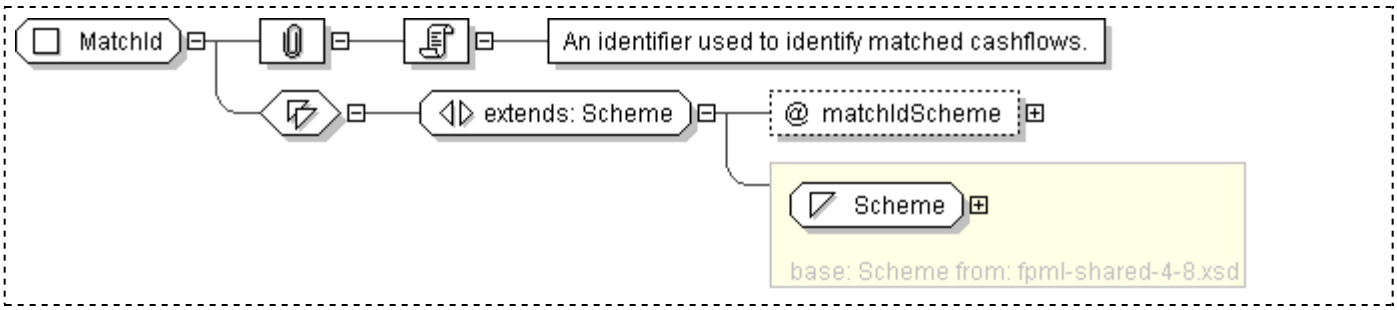
Super-types:	Scheme < MatchId (by extension)
Sub-types:	None

Name	MatchId
Used by (from the same schema document)	Complex Type CancelTradeCashflows , Complex Type PositionProposedMatch , Complex Type TradeCashflowsAsserted , Complex Type TradeCashflowsProposedMatch
Abstract	no
Documentation	An identifier used to identify matched cashflows.

XML Instance Representation

```
<...
 matchIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MatchId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="matchIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentDetails

[Table of contents]

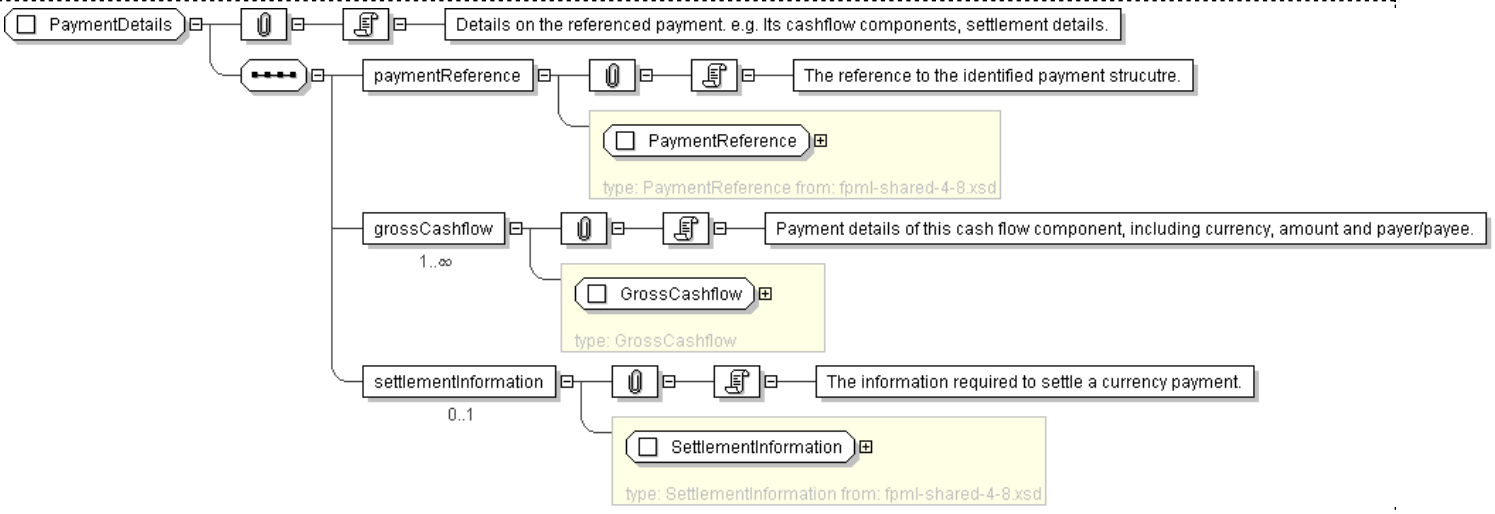
Super-types:	None
Sub-types:	None

Name	PaymentDetails
Abstract	no
Documentation	Details on the referenced payment. e.g. Its cashflow components, settlement details.

XML Instance Representation

```
<...>
  <paymentReference> PaymentReference </paymentReference> [1]
  'The reference to the identified payment strucutre.'

  <grossCashflow> GrossCashflow </grossCashflow> [1..*]
  'Payment details of this cash flow component, including currency, amount and payer/payee.'

  <settlementInformation> SettlementInformation </settlementInformation> [0..1]
  'The information required to settle a currency payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentDetails">
  <xsd:sequence>
    <xsd:element name="paymentReference" type=" PaymentReference "/>
    <xsd:element name="grossCashflow" type=" GrossCashflow " maxOccurs="unbounded"/>
    <xsd:element name="settlementInformation" type=" SettlementInformation " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentId

[Table of contents]

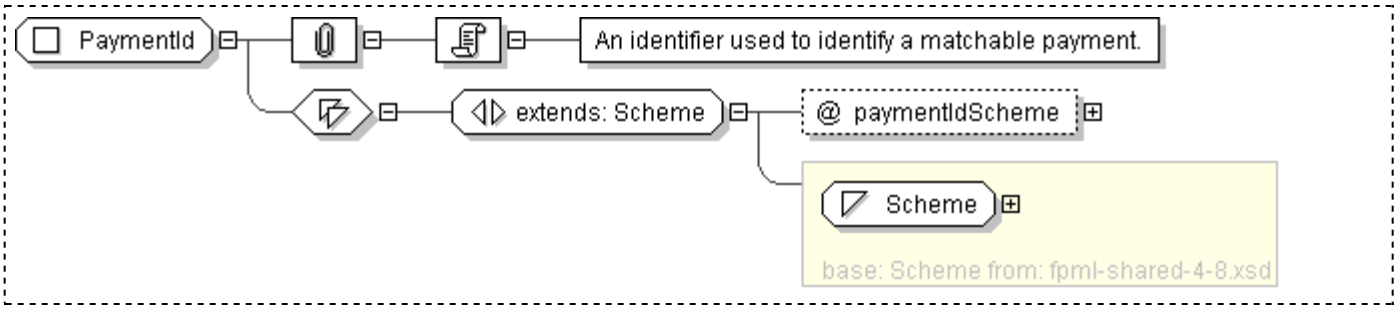
Super-types:	Scheme < PaymentId (by extension)
Sub-types:	None

Name	PaymentId
Used by (from the same schema document)	Complex Type PaymentMatching
Abstract	no
Documentation	An identifier used to identify a matchable payment.

XML Instance Representation

```
<...
 paymentIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="paymentIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentMatching

[Table of contents]

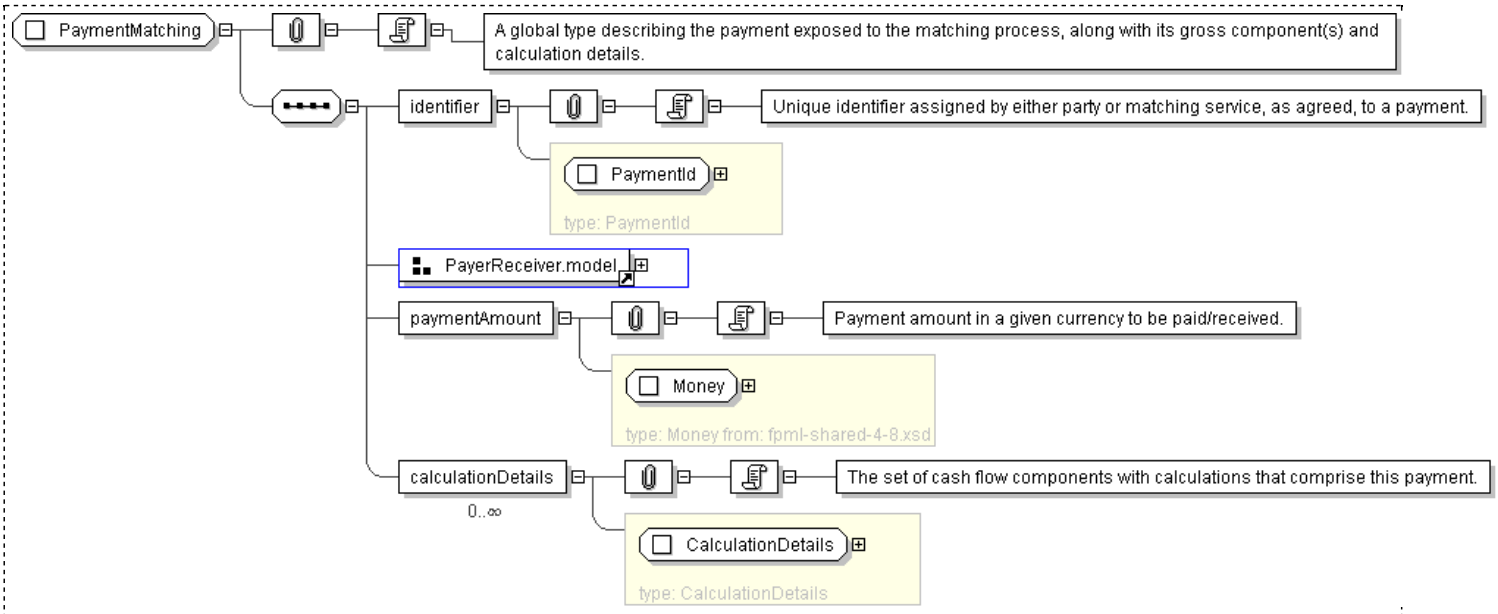
Super-types:	None
Sub-types:	None

Name	PaymentMatching
Used by (from the same schema document)	Model Group TradeCashflows.model
Abstract	no
Documentation	A global type describing the payment exposed to the matching process, along with its gross component(s) and calculation details.

XML Instance Representation

```
<...>
  <identifier> PaymentId </identifier> [1]
  'Unique identifier assigned by either party or matching service, as agreed, to a payment.'

  <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
  'A reference to the party responsible for making the payments defined by this structure.'

  <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
  'A reference to the party that receives the payments corresponding to this structure.'

  <paymentAmount> Money </paymentAmount> [1]
  'Payment amount in a given currency to be paid/received.'

  <calculationDetails> CalculationDetails </calculationDetails> [0..*]
  'The set of cash flow components with calculations that comprise this payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentMatching">
  <xsd:sequence>
    <xsd:element name="identifier" type=" PaymentId "/>
    <xsd:group ref=" PayerReceiver.model "/>
    <xsd:element name="paymentAmount" type=" Money "/>
    <xsd:element name="calculationDetails" type=" CalculationDetails " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PortfolioDefinition

[Table of contents]

Super-types: None

Sub-types:

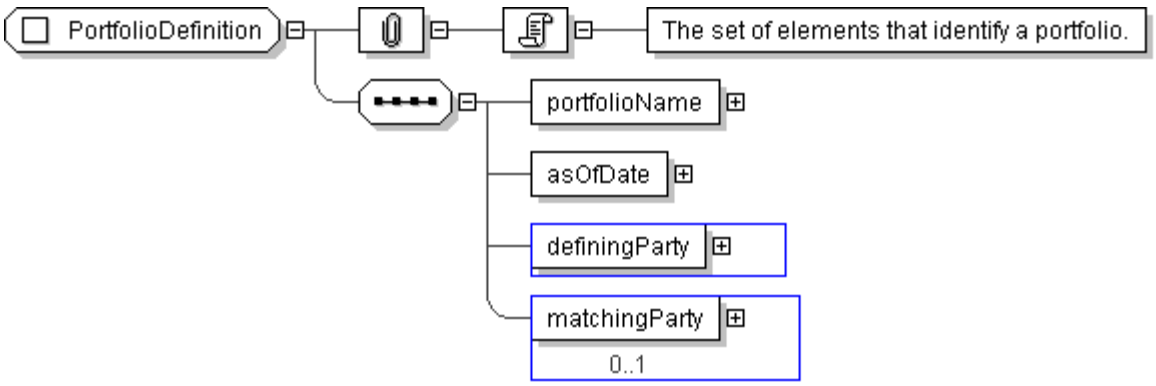
- InitialPortfolioDefinition (by extension)

Name	PortfolioDefinition
Used by (from the same schema document)	Complex Type PositionsAcknowledged , Complex Type PositionsMatchResults
Abstract	no
Documentation	The set of elements that identify a portfolio.

XML Instance Representation

```
<...>
   <portfolioName> xsd:normalizedString </portfolioName> [1]
   <asOfDate> xsd:date </asOfDate> [1]
   <definingParty> PartyReference </definingParty> [1]
   <matchingParty> PartyReference </matchingParty> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PortfolioDefinition">
   <xsd:sequence>
      <xsd:element name="portfolioName" type=" xsd:normalizedString "/>
      <xsd:element name="asOfDate" type=" xsd:date "/>
      <xsd:element name="definingParty" type=" PartyReference "/>
      <xsd:element name="matchingParty" type=" PartyReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionMatchResult

[Table of contents]

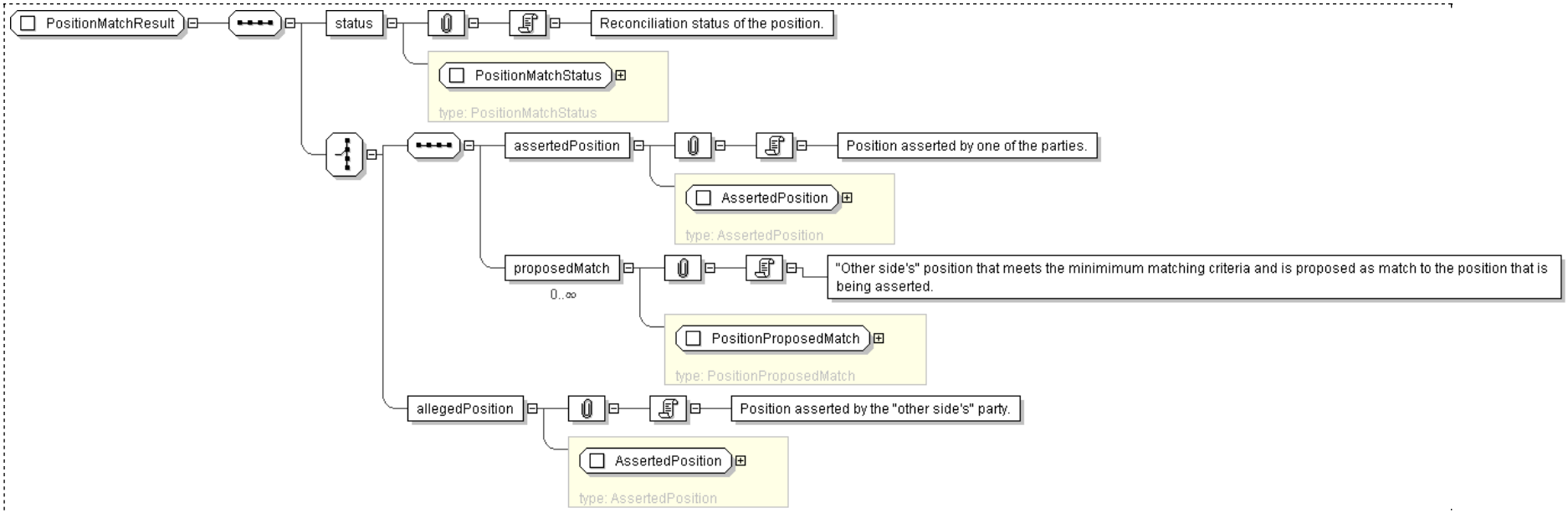
Super-types:	None
Sub-types:	None

Name	PositionMatchResult
Used by (from the same schema document)	Complex Type PositionsMatchResults
Abstract	no

XML Instance Representation

```
<...>
  <status> PositionMatchStatus </status> [1]
   'Reconciliation status of the position.'

  Start Choice [1]
    <assertedPosition> AssertedPosition </assertedPosition> [1]
     'Position asserted by one of the parties.'

    <proposedMatch> PositionProposedMatch </proposedMatch> [0..*]
     '"Other side\'s" position that meets the minimimum matching criteria and is proposed as match to the position that is being asserted.'

    <allegedPosition> AssertedPosition </allegedPosition> [1]
     'Position asserted by the \"other side\'s\" party.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionMatchResult">
  <xsd:sequence>
    <xsd:element name="status" type=" PositionMatchStatus "/>
    <xsd:choice>
      <xsd:sequence>
```

```
          <xsd:element name="assertedPosition" type=" AssertedPosition "/>
          <xsd:element name="proposedMatch" type=" PositionProposedMatch " minOccurs="0" maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:element name="allegedPosition" type=" AssertedPosition "/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionMatchStatus

[Table of contents]

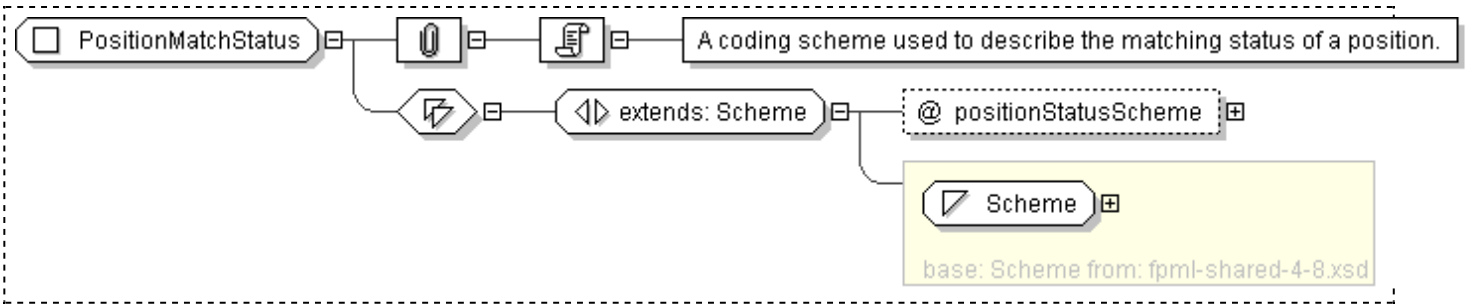
Super-types:	Scheme < PositionMatchStatus (by extension)
Sub-types:	None

Name	PositionMatchStatus
Used by (from the same schema document)	Complex Type PositionMatchResult
Abstract	no
Documentation	A coding scheme used to describe the matching status of a position.

XML Instance Representation

```
<...
 positionStatusScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionMatchStatus">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="positionStatusScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/position-status"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionProposedMatch

[Table of contents]

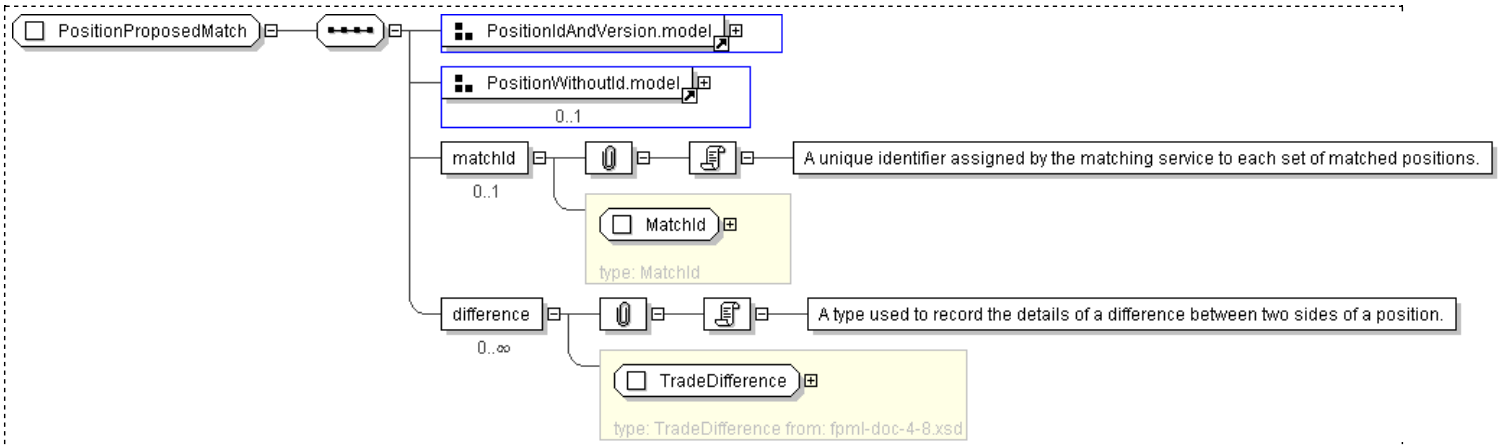
Super-types:	None
Sub-types:	None

Name	PositionProposedMatch
Used by (from the same schema document)	Complex Type PositionMatchResult
Abstract	no

XML Instance Representation

```
<...>
   <positionId> PositionId </positionId> [1]
   'A version-independent identifier for the position, possibly based on trade identifier.'

   <version> xsd:positiveInteger </version> [0..1]
   'A version identifier. Version identifiers must be ascending, i.e. higher numbers imply newer versions. There is
   no requirement that version identifiers for a position be sequential or small, so for example timestamp-based
   version identifiers could be used.'

   Start Group: PositionWithoutId.model [0..1]
      <reportingRoles> ReportingRoles </reportingRoles> [0..1]
      'Information about the roles of the parties with respect to reporting the positions.'

      <constituent> PositionConstituent </constituent> [1]
      'The components that create this position.'

      <scheduledDate> ScheduledDate </scheduledDate> [0..*]
      'Position level schedule date, such as final payment dates, in a simple and flexible format.'

      <valuation> AssetValuation </valuation> [0..*]
      'Valuation reported for the position, such as NPV or accrued interest. The asset/object references in the
      valuations should refer to the deal or components of the deal in the position, e.g. legs, streams, or
      underlyers.'

   End Group: PositionWithoutId.model
   <matchId> MatchId </matchId> [0..1]
   'A unique identifier assigned by the matching service to each set of matched positions.'

   <difference> TradeDifference </difference> [0..*]
   'A type used to record the details of a difference between two sides of a position.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionProposedMatch">
   <xsd:sequence>
      <xsd:group ref=" PositionIdAndVersion.model "/>
      <xsd:group ref=" PositionWithoutId.model " minOccurs="0"/>
      <xsd:element name="matchId" type=" MatchId " minOccurs="0"/>
      <xsd:element name="difference" type=" TradeDifference " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionReference

[Table of contents]

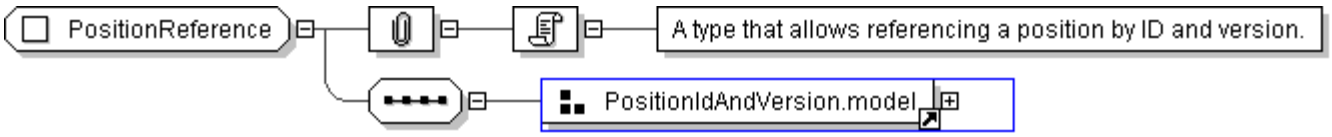
Super-types:	None
Sub-types:	None

Name	PositionReference
Used by (from the same schema document)	Complex Type DefinePosition , Complex Type PositionsAcknowledged , Complex Type PositionsAcknowledged , Complex Type PositionsAsserted
Abstract	no
Documentation	A type that allows referencing a position by ID and version.

XML Instance Representation

```
<...>
   <positionId> PositionId </positionId> [1]
   'A version-independent identifier for the position, possibly based on trade
   identifier.'

   <version> xsd:positiveInteger </version> [0..1]
   'A version identifier. Version identifiers must be ascending, i.e. higher
   numbers imply newer versions. There is no requirement that version identifiers
   for a position be sequential or small, so for example timestamp-based version
   identifiers could be used.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionReference">
   <xsd:sequence>
      <xsd:group ref=" PositionIdAndVersion.model "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionsAcknowledged

[Table of contents]

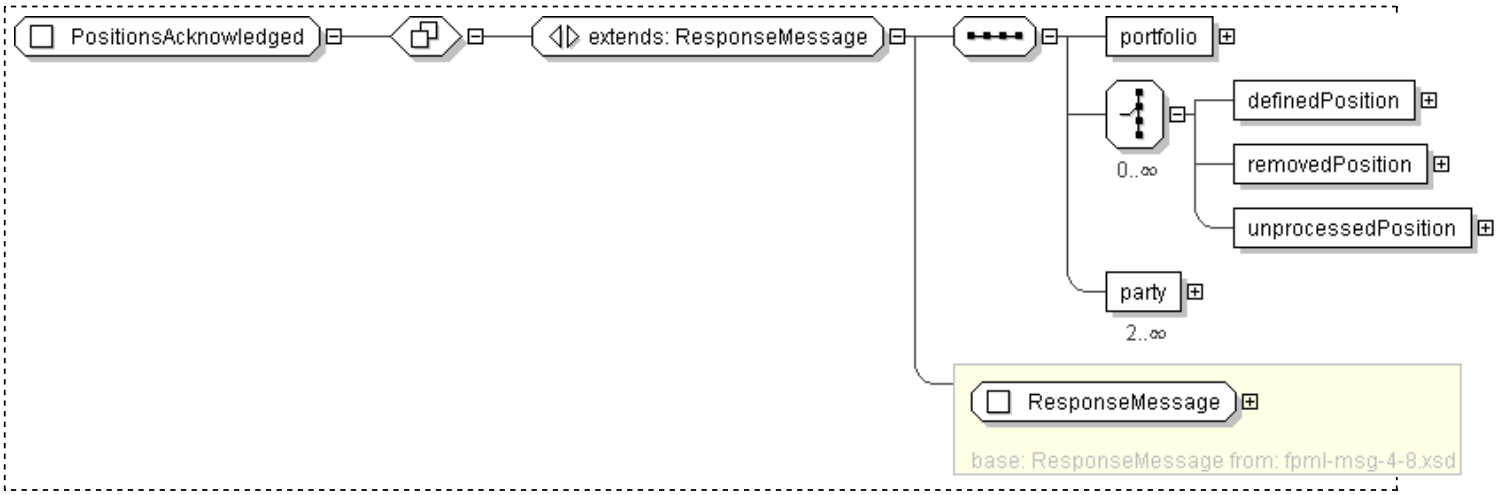
Super-types:	ResponseMessage < PositionsAcknowledged (by extension)
Sub-types:	None

Name	PositionsAcknowledged
Abstract	no

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <portfolio> PortfolioDefinition </portfolio> [1]
   'Contains the portfolio definition.'

   Start Choice [0..*]
      <definedPosition> PositionReference </definedPosition> [1]
      <removedPosition> PositionReference </removedPosition> [1]
      <unprocessedPosition> UnprocessedPosition </unprocessedPosition> [1]
   End Choice
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionsAcknowledged">
   <xsd:complexContent>
```

```
      <xsd:extension base=" ResponseMessage ">
        <xsd:sequence>
          <xsd:element name="portfolio" type=" PortfolioDefinition "/>
          <xsd:choice minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="definedPosition" type=" PositionReference "/>
            <xsd:element name="removedPosition" type=" PositionReference "/>
            <xsd:element name="unprocessedPosition" type=" UnprocessedPosition "/>
          </xsd:choice>
          <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionsAsserted

[Table of contents]

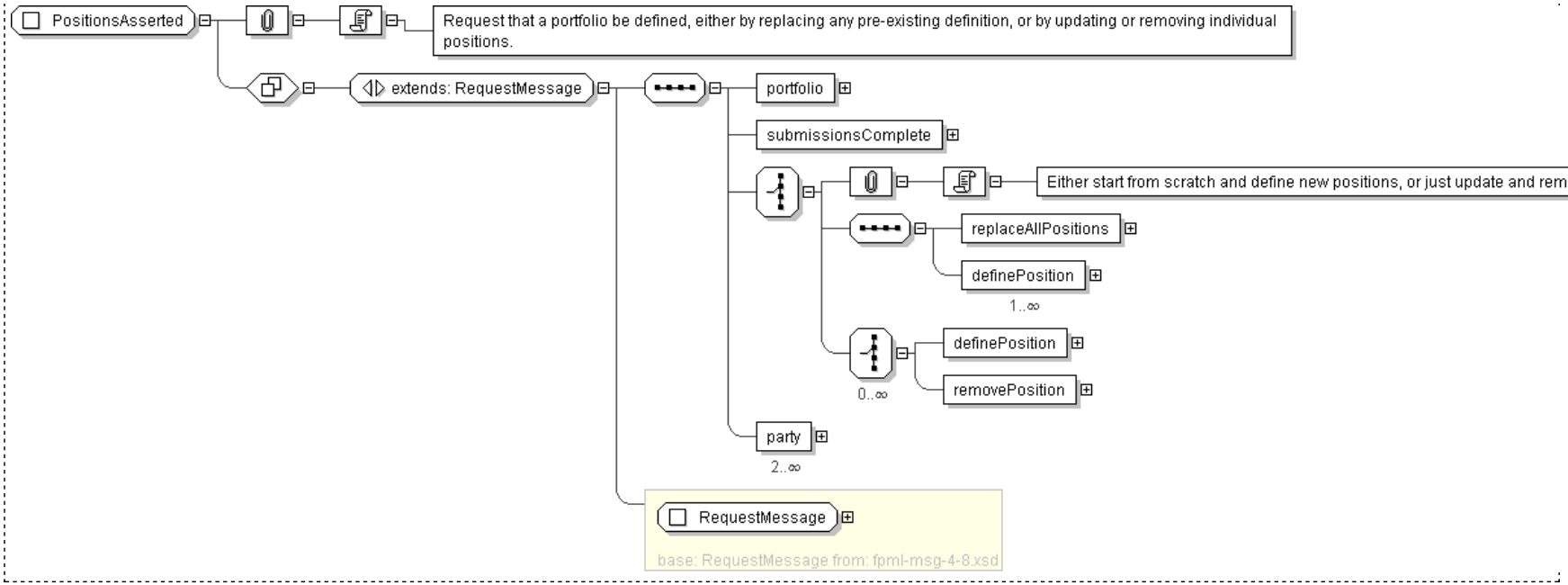
Super-types:	RequestMessage < PositionsAsserted (by extension)
Sub-types:	None

Name	PositionsAsserted
Abstract	no
Documentation	Request that a portfolio be defined, either by replacing any pre-existing definition, or by updating or removing individual positions.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define
 the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML
 parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a
 change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <portfolio> InitialPortfolioDefinition </portfolio> [1]
   'Contains the portfolio definition.'

   <submissionsComplete> xsd:boolean </submissionsComplete> [1]
   'Indicates whether all portfolio updates have been submitted for this as-of date'

   Start Choice [1]
 'Either start from scratch and define new positions, or just update and remove positions'


      <replaceAllPositions> Empty </replaceAllPositions> [1]
      'Indicates that this message replaces all previous positions for this portfolio.'

      <definePosition> DefinePosition </definePosition> [1..*]
      'Used to specify a new position.'

      Start Choice [0..*]
         <definePosition> DefinePosition </definePosition> [1]
         'Used to specify a position, whether it is a new or updated position.'

         <removePosition> PositionReference </removePosition> [1]
         'Used to remove a position from the portfolio.'

      End Choice
   End Choice
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionsAsserted">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="portfolio" type=" InitialPortfolioDefinition "/>
        <xsd:element name="submissionsComplete" type=" xsd:boolean "/>
        <xsd:choice>
          <xsd:sequence>
            <xsd:element name="replaceAllPositions" type=" Empty "/>
            <xsd:element name="definePosition" type=" DefinePosition " maxOccurs="unbounded"/>
          </xsd:sequence>
          <xsd:choice minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="definePosition" type=" DefinePosition "/>
            <xsd:element name="removePosition" type=" PositionReference "/>
          </xsd:choice>
        </xsd:choice>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionsMatchResults

[Table of contents]

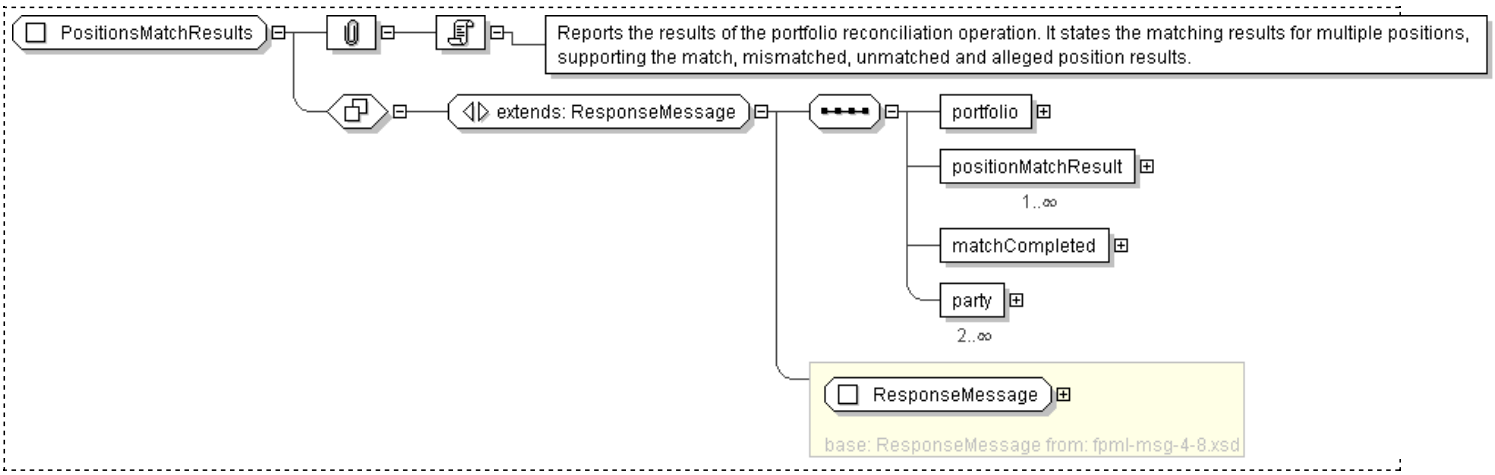
Super-types:	ResponseMessage < PositionsMatchResults (by extension)
Sub-types:	None

Name	PositionsMatchResults
Abstract	no
Documentation	Reports the results of the portfolio reconciliation operation. It states the matching results for multiple positions, supporting the match, mismatched, unmatched and alleged position results.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <portfolio> PortfolioDefinition </portfolio> [1]
   'Contains the portfolio definition.'

   <positionMatchResult> PositionMatchResult </positionMatchResult> [1..*]
   <matchCompleted> xsd:boolean </matchCompleted> [1]
   'Flag indicating whether the Matching Service has finished sending all matching results.'

   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionsMatchResults">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="portfolio" type=" PortfolioDefinition "/>
        <xsd:element name="positionMatchResult" type=" PositionMatchResult " maxOccurs="unbounded"/>
        <xsd:element name="matchCompleted" type=" xsd:boolean "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestPortfolio

[Table of contents]

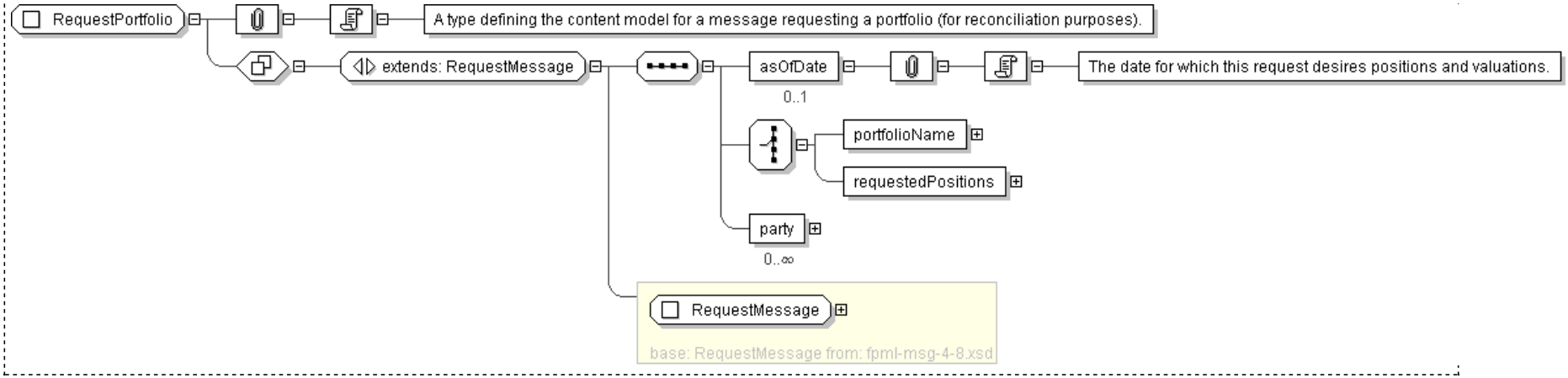
Super-types:	RequestMessage < RequestPortfolio (by extension)
Sub-types:	None

Name	RequestPortfolio
Abstract	no
Documentation	A type defining the content model for a message requesting a portfolio (for reconciliation purposes).

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used
 to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by
 the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time
 FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
 actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <asOfDate> ... </asOfDate> [0..1]
   'The date for which this request desires positions and valuations.'

   Start Choice [1]
      <portfolioName> xsd:normalizedString </portfolioName> [1]
      'The name of the portfolio that is requested.'

      <requestedPositions> RequestedPositions </requestedPositions> [1]
      'The name of the data set (portfolio, product type, etc.) that this request corresponds to. Describes the desired report.'

   End Choice
   <party> Party </party> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestPortfolio">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="asOfDate" minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="portfolioName" type=" xsd:normalizedString "/>
          <xsd:element name="requestedPositions" type=" RequestedPositions "/>
        </xsd:choice>
        <xsd:element name="party" type=" Party " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StepReference

[Table of contents]

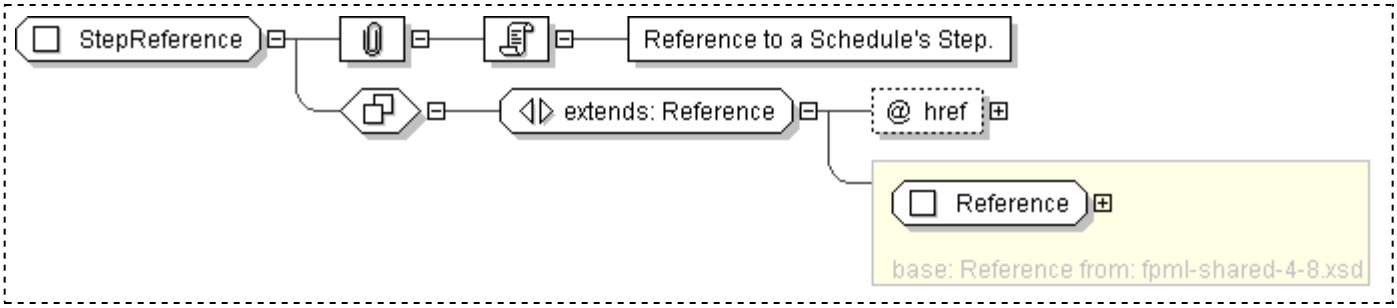
Super-types:	Reference < StepReference (by extension)
Sub-types:	None

Name	StepReference
Used by (from the same schema document)	Complex Type CashflowCalculationPeriod
Abstract	no
Documentation	Reference to a Schedule's Step.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StepReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Step"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCashflowsAsserted

[Table of contents]

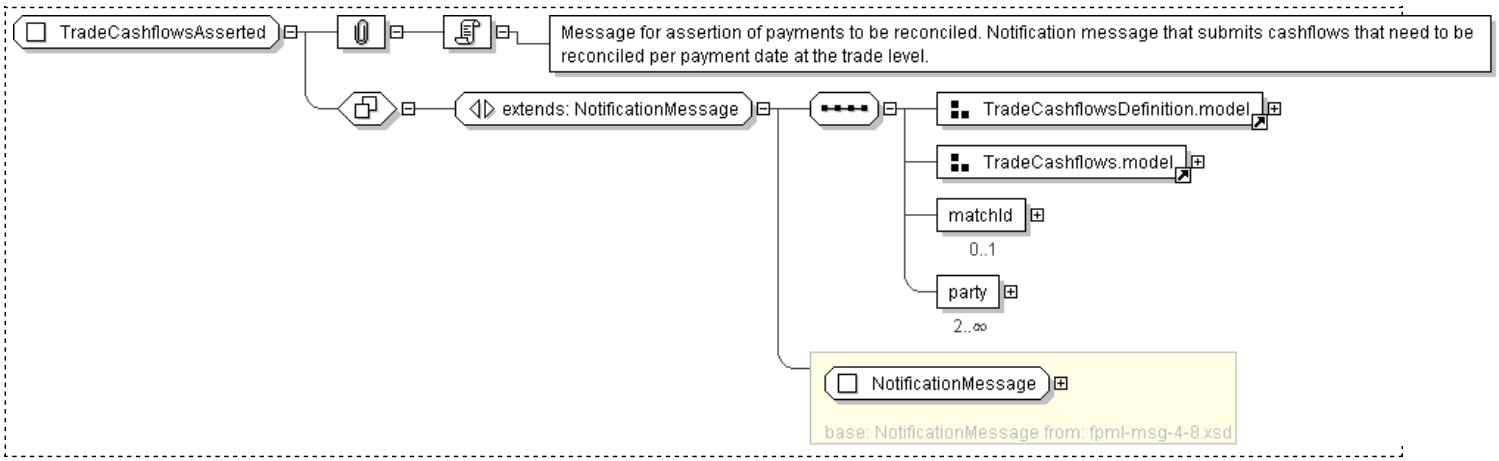
Super-types:	NotificationMessage < TradeCashflowsAsserted (by extension)
Sub-types:	None

Name	TradeCashflowsAsserted
Abstract	no
Documentation	Message for assertion of payments to be reconciled. Notification message that submits cashflows that need to be reconciled per payment date at the trade level.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of
 the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the
 build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples
 within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between
 releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the
 same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <asOfDate> xsd:dateTime </asOfDate> [0..1]
   'The date and time at which the set of cashflows was defined.'

   <tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
   'Unique identifier assigned by the party asserting the set of cashflows to be reconciled.'

   <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
   'Structure that holds reference to the trade through the tradeId and optionally some trade-specific elements
   for identifying the trade in the case of trades that have not been negotiated through electronic platforms and
   for which the counterparty\'s trade ID has not been captured.'

   <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
   'The adjusted date in which the payments are being paid/received.'

   <payment> PaymentMatching </payment> [1..*]
   'Specifies the payment that is exposed to the matching process. Usually there will be a single payment but for
   cross-currency swaps a different payment per currency shall be provided.'

   <matchId> MatchId </matchId> [0..1]
   'A unique identifier assigned by either party, or matching service, as agreed, to each set of matched
   cashflows.'

   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCashflowsAsserted">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:group ref=" TradeCashflowsDefinition.model "/>
            <xsd:group ref=" TradeCashflows.model "/>
            <xsd:element name="matchId" type=" MatchId " minOccurs="0"/>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCashflowsId

[Table of contents]

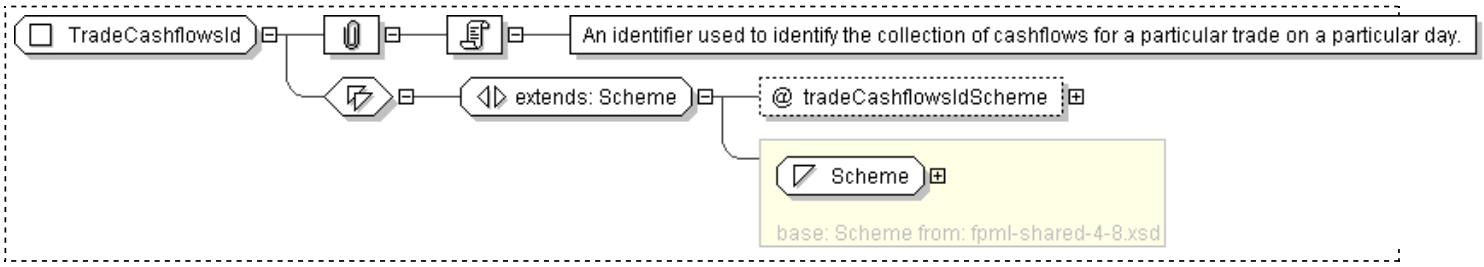
Super-types:	Scheme < TradeCashflowsId (by extension)
Sub-types:	None

Name	TradeCashflowsId
Used by (from the same schema document)	Model Group IdAndTradeCashflows.model , Model Group TradeCashflowsDefinition.model
Abstract	no
Documentation	An identifier used to identify the collection of cashflows for a particular trade on a particular day.

XML Instance Representation

```
<...
 tradeCashflowsIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCashflowsId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="tradeCashflowsIdScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCashflowsMatchResult

[Table of contents]

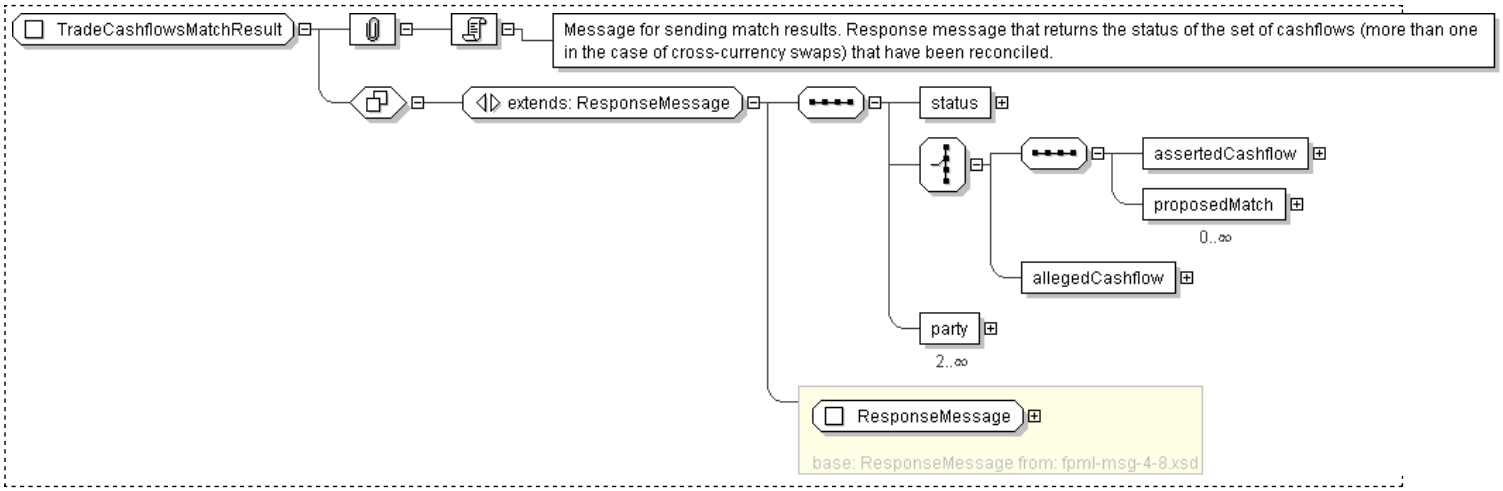
Super-types:	ResponseMessage < TradeCashflowsMatchResult (by extension)
Sub-types:	None

Name	TradeCashflowsMatchResult
Abstract	no
Documentation	Message for sending match results. Response message that returns the status of the set of cashflows (more than one in the case of cross-currency swaps) that have been reconciled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the
 schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it
 is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of
 the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version
 (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a
 version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <status> TradeCashflowsStatus </status> [1]
   'Reconciliation status of the set of cashflows.'

   Start Choice [1]
      <assertedCashflow> AssertedCashflow </assertedCashflow> [1]
      'Cashflow (or set of cashflows for cross-currency swap) asserted by one of the parties.'

      <proposedMatch> TradeCashflowsProposedMatch </proposedMatch> [0..*]
      '"Other side\'s" cashflow that meets the minimimum matching criteria and is proposed as match to the cashflow that
      is being asserted.'

      <allegedCashflow> AllegedCashflow </allegedCashflow> [1]
      'Cashflow (or set of cashflows for cross-currency swap) asserted by the \"other side\'s\" party.'

   End Choice
   <party> Party </party> [2..*]
   'One party element for each of the principal parties and any other party that is referenced.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCashflowsMatchResult">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="status" type=" TradeCashflowsStatus "/>
        <xsd:choice>
          <xsd:sequence>
            <xsd:element name="assertedCashflow" type=" AssertedCashflow "/>
```

```
                <xsd:element name="proposedMatch" type=" TradeCashflowsProposedMatch " minOccurs="0" maxOccurs="unbounded"/>
              </xsd:sequence>
              <xsd:element name="allegedCashflow" type=" AllegedCashflow "/>
            </xsd:choice>
            <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCashflowsProposedMatch

[Table of contents]

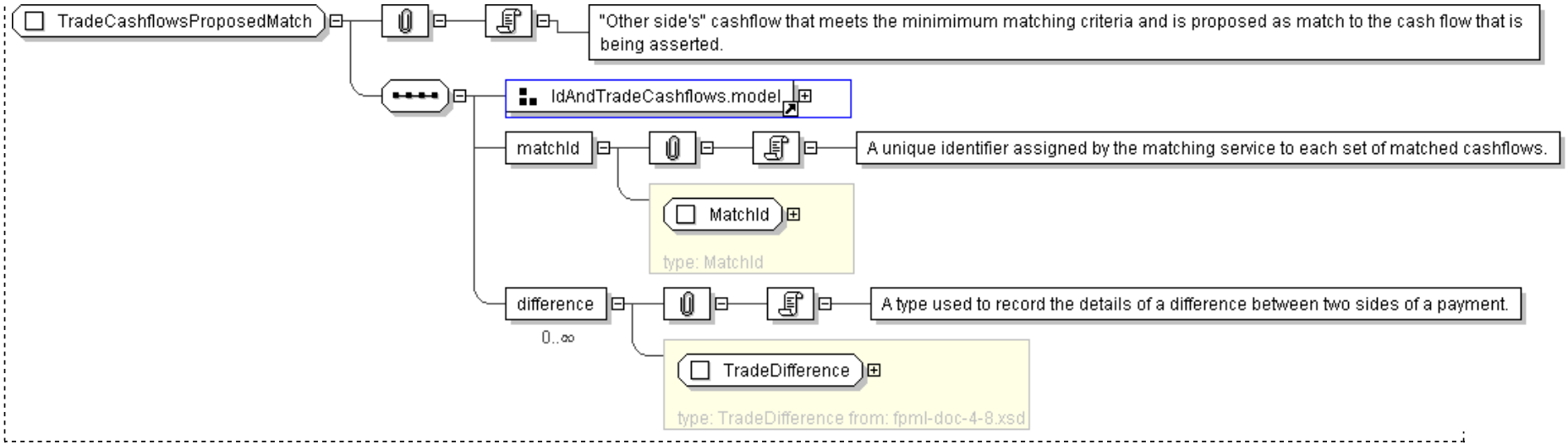
Super-types:	None
Sub-types:	None

Name	TradeCashflowsProposedMatch
Used by (from the same schema document)	Complex Type TradeCashflowsMatchResult
Abstract	no
Documentation	"Other side's" cashflow that meets the minimimum matching criteria and is proposed as match to the cash flow that is being asserted.

XML Instance Representation

```
<...>
   <tradeCashflowsId> TradeCashflowsId </tradeCashflowsId> [1]
   'Unique identifier assigned by either party to a set of cashflows.'

   Start Group: TradeCashflows.model [0..1]
      <tradeIdentifyingItems> TradeIdentifyingItems </tradeIdentifyingItems> [1]
      'Structure that holds reference to the trade through the tradeId and optionally some trade-specific elements for
      identifying the trade in the case of trades that have not been negotiated through electronic platforms and for which
      the counterparty\'s trade ID has not been captured.'

      <adjustedPaymentDate> xsd:date </adjustedPaymentDate> [1]
      'The adjusted date in which the payments are being paid/received.'

      <payment> PaymentMatching </payment> [1..*]
      'Specifies the payment that is exposed to the matching process. Usually there will be a single payment but for cross-
      currency swaps a different payment per currency shall be provided.'

   End Group: TradeCashflows.model
   <matchId> MatchId </matchId> [1]
   'A unique identifier assigned by the matching service to each set of matched cashflows.'

   <difference> TradeDifference </difference> [0..*]
   'A type used to record the details of a difference between two sides of a payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCashflowsProposedMatch">
  <xsd:sequence>
    <xsd:group ref=" IdAndTradeCashflows.model "/>
    <xsd:element name="matchId" type=" MatchId "/>
    <xsd:element name="difference" type=" TradeDifference " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCashflowsStatus

[Table of contents]

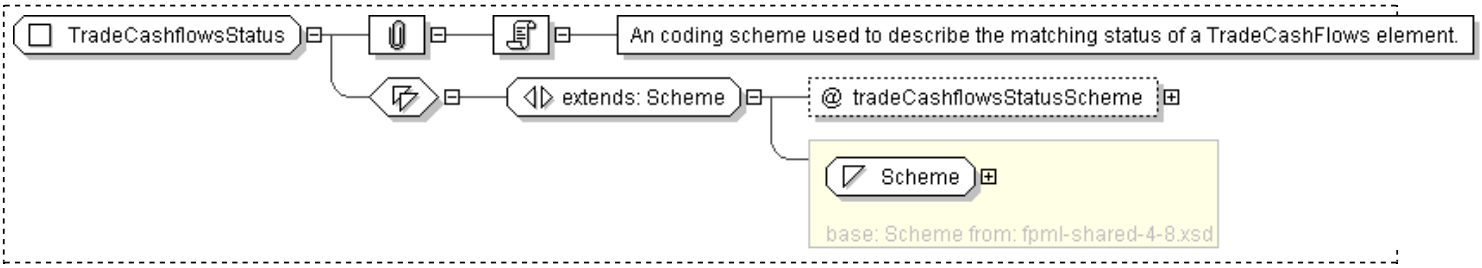
Super-types:	Scheme < TradeCashflowsStatus (by extension)
Sub-types:	None

Name	TradeCashflowsStatus
Used by (from the same schema document)	Complex Type TradeCashflowsMatchResult
Abstract	no
Documentation	An coding scheme used to describe the matching status of a TradeCashFlows element.

XML Instance Representation

```
<...
 tradeCashflowsStatusScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCashflowsStatus">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="tradeCashflowsStatusScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/trade-cashflows-status"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeDetails

[Table of contents]

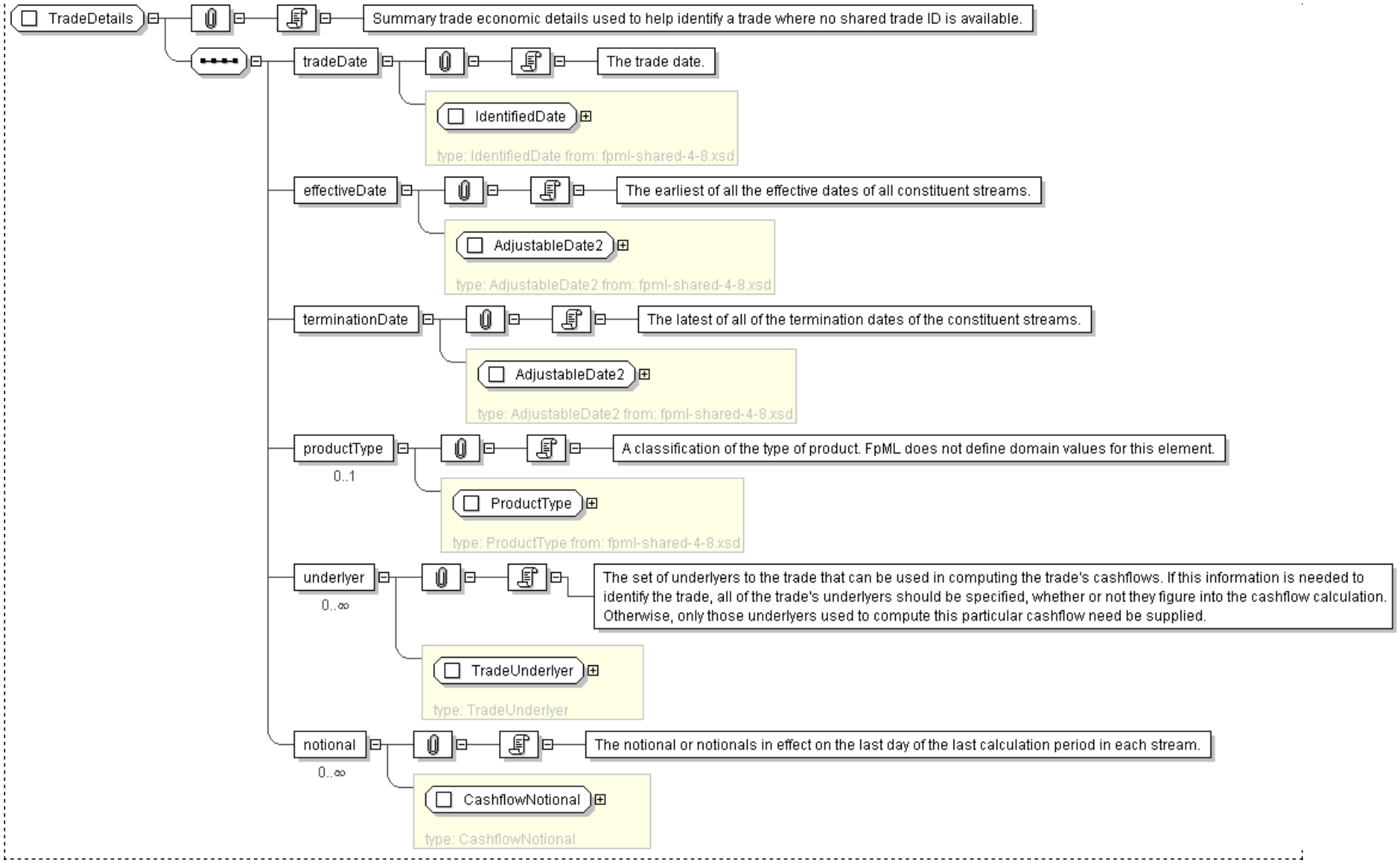
Super-types:	None
Sub-types:	None

Name	TradeDetails
Used by (from the same schema document)	Complex Type TradeIdentifyingItems
Abstract	no
Documentation	Summary trade economic details used to help identify a trade where no shared trade ID is available.

XML Instance Representation

```
<...>
   <tradeDate> IdentifiedDate </tradeDate> [1]
   'The trade date.'

   <effectiveDate> AdjustableDate2 </effectiveDate> [1]
   'The earliest of all the effective dates of all constituent streams.'

   <terminationDate> AdjustableDate2 </terminationDate> [1]
   'The latest of all of the termination dates of the constituent streams.'

   <productType> ProductType </productType> [0..1]
   'A classification of the type of product. FpML does not define domain values for this element.'

   <underlyer> TradeUnderlyer </underlyer> [0..*]
   'The set of underlyers to the trade that can be used in computing the trade\'s cashflows. If this information is needed to
   identify the trade, all of the trade\'s underlyers should be specified, whether or not they figure into the cashflow calculation.
   Otherwise, only those underlyers used to compute this particular cashflow need be supplied.'

   <notional> CashflowNotional </notional> [0..*]
   'The notional or notionals in effect on the last day of the last calculation period in each stream.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeDetails">
  <xsd:sequence>
    <xsd:element name="tradeDate" type=" IdentifiedDate "/>
    <xsd:element name="effectiveDate" type=" AdjustableDate2 "/>
    <xsd:element name="terminationDate" type=" AdjustableDate2 "/>
    <xsd:element name="productType" type=" ProductType " minOccurs="0"/>
    <xsd:element name="underlyer" type=" TradeUnderlyer " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="notional" type=" CashflowNotional " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeIdentifyingItems

[Table of contents]

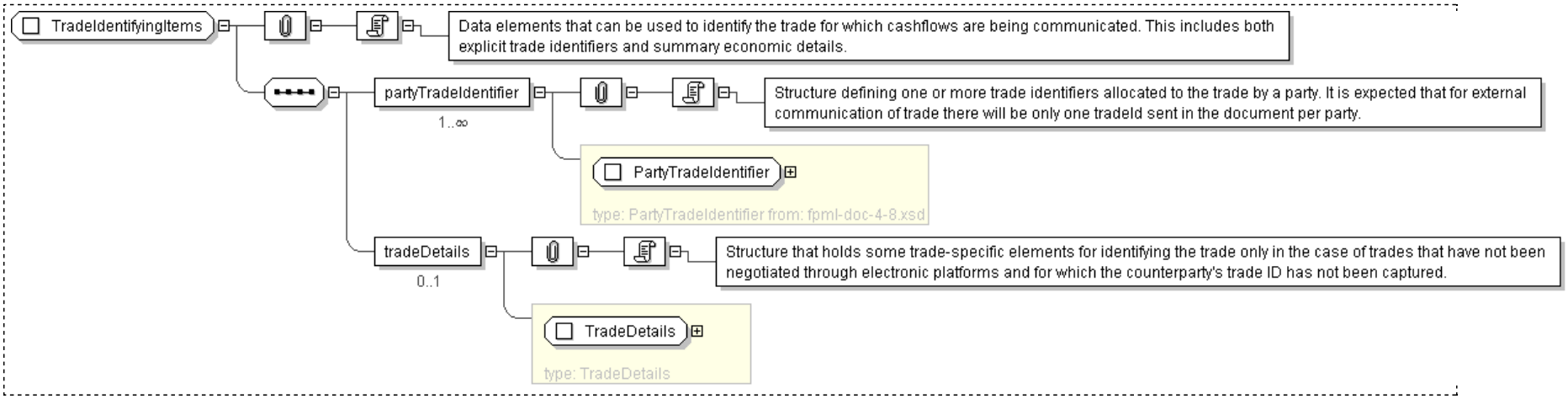
Super-types:	None
Sub-types:	None

Name	TradeIdentifyingItems
Used by (from the same schema document)	Model Group TradeCashflows.model
Abstract	no
Documentation	Data elements that can be used to identify the trade for which cashflows are being communicated. This includes both explicit trade identifiers and summary economic details.

XML Instance Representation

```
<...>
   <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1..*]
   'Structure defining one or more trade identifiers allocated to the trade by a party. It is expected that for external communication of
   trade there will be only one tradeId sent in the document per party.'

   <tradeDetails> TradeDetails </tradeDetails> [0..1]
   'Structure that holds some trade-specific elements for identifying the trade only in the case of trades that have not been negotiated
   through electronic platforms and for which the counterparty\'s trade ID has not been captured.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeIdentifyingItems">
   <xsd:sequence>
      <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier " maxOccurs="unbounded"/>
      <xsd:element name="tradeDetails" type=" TradeDetails " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeUnderlyer

[Table of contents]

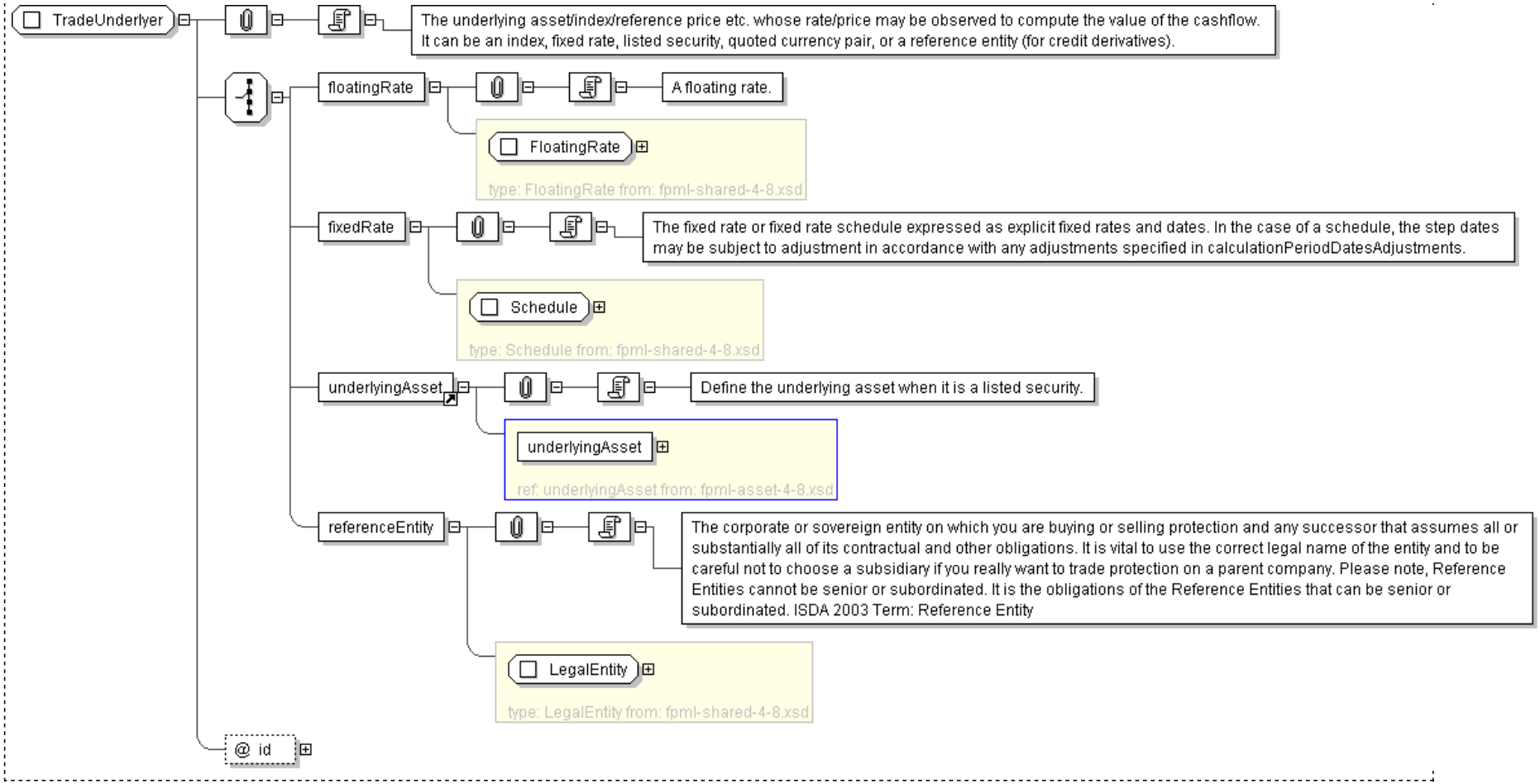
Super-types:	None
Sub-types:	None

Name	TradeUnderlyer
Used by (from the same schema document)	Complex Type CashflowCalculationElements , Complex Type TradeDetails
Abstract	no
Documentation	The underlying asset/index/reference price etc. whose rate/price may be observed to compute the value of the cashflow. It can be an index, fixed rate, listed security, quoted currency pair, or a reference entity (for credit derivatives).

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <floatingRate> FloatingRate </floatingRate> [1]
       'A floating rate.'

      <fixedRate> Schedule </fixedRate> [1]
       'The fixed rate or fixed rate schedule expressed as explicit fixed rates and dates. In the case of a schedule, the step dates
       may be subject to adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments.'

      <underlyingAsset> ... </underlyingAsset> [1]
       'Define the underlying asset when it is a listed security.'

      <referenceEntity> LegalEntity </referenceEntity> [1]
       'The corporate or sovereign entity on which you are buying or selling protection and any successor that assumes all or
       substantially all of its contractual and other obligations. It is vital to use the correct legal name of the entity and to be
       careful not to choose a subsidiary if you really want to trade protection on a parent company. Please note, Reference Entities
       cannot be senior or subordinated. It is the obligations of the Reference Entities that can be senior or subordinated. ISDA 2003
       Term: Reference Entity'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeUnderlyer">
  <xsd:choice>
    <xsd:element name="floatingRate" type=" FloatingRate "/>
    <xsd:element name="fixedRate" type=" Schedule "/>
    <xsd:element ref=" underlyingAsset "/>
    <xsd:element name="referenceEntity" type=" LegalEntity "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeUnderlyerReference

[Table of contents]

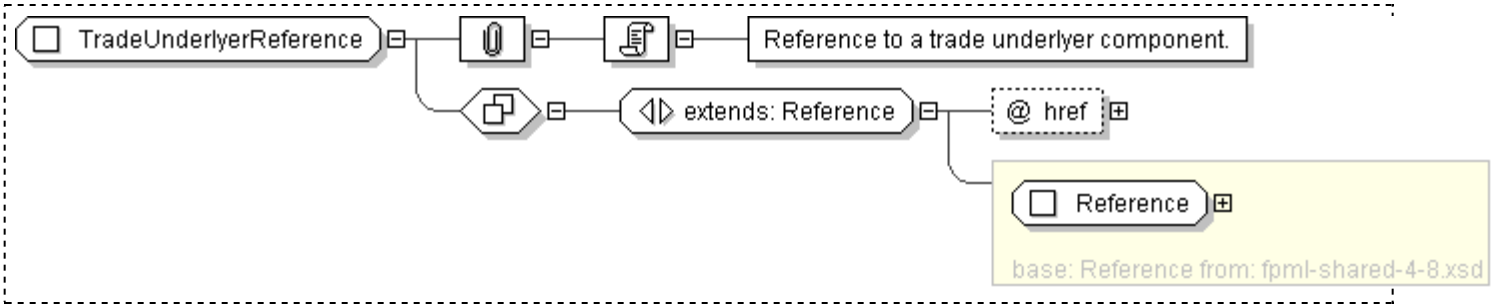
Super-types:	Reference < TradeUnderlyerReference (by extension)
Sub-types:	None

Name	TradeUnderlyerReference
Used by (from the same schema document)	Complex Type CashflowObservation , Complex Type UnderlyerReferenceUnits
Abstract	no
Documentation	Reference to a trade underlyer component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeUnderlyerReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="TradeUnderlyer"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: UnderlyerReferenceUnits

[Table of contents]

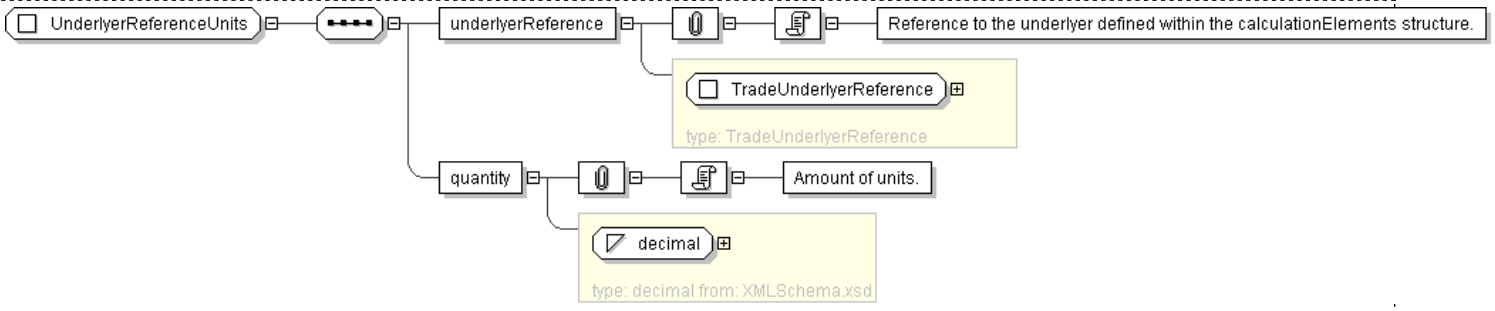
Super-types:	None
Sub-types:	None

Name	UnderlyerReferenceUnits
Used by (from the same schema document)	Complex Type CashflowCalculationElements
Abstract	no

XML Instance Representation

```
<...>
   <underlyerReference> TradeUnderlyerReference </underlyerReference> [1]
   'Reference to the underlyer defined within the calculationElements structure.'

   <quantity> xsd:decimal </quantity> [1]
   'Amount of units.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="UnderlyerReferenceUnits">
   <xsd:sequence>
      <xsd:element name="underlyerReference" type=" TradeUnderlyerReference "/>
      <xsd:element name="quantity" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: UnprocessedPosition

[Table of contents]

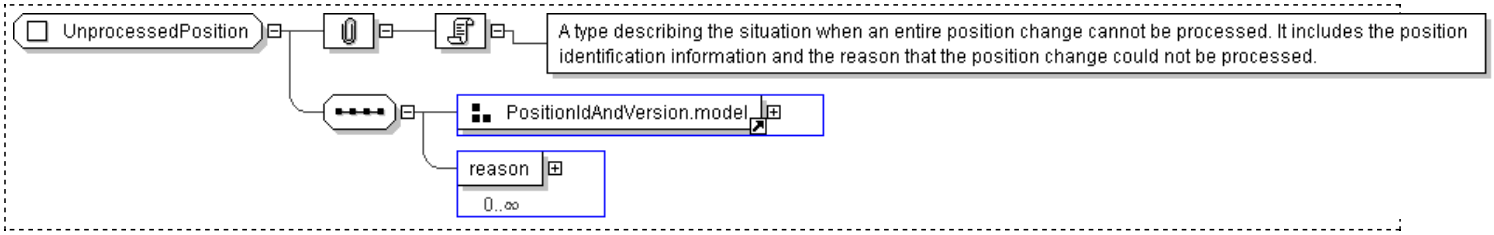
Super-types:	None
Sub-types:	None

Name	UnprocessedPosition
Used by (from the same schema document)	Complex Type PositionsAcknowledged
Abstract	no
Documentation	A type describing the situation when an entire position change cannot be processed. It includes the position identification information and the reason that the position change could not be processed.

XML Instance Representation

```
<...>
  <positionId> PositionId </positionId> [1]
  'A version-independent identifier for the position, possibly based on trade identifier.'

  <version> xsd:positiveInteger </version> [0..1]
  'A version identifier. Version identifiers must be ascending, i.e. higher numbers imply newer versions.
  There is no requirement that version identifiers for a position be sequential or small, so for example
  timestamp-based version identifiers could be used.'

  <reason> Reason </reason> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="UnprocessedPosition">
  <xsd:sequence>
    <xsd:group ref=" PositionIdAndVersion.model "/>
    <xsd:element name="reason" type=" Reason " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: portfolio
  - Element: queryPortfolio
- Global Definitions
  - Complex Type: PortfolioValuationItem
  - Complex Type: PositionReport
  - Complex Type: RequestPositionReport
  - Complex Type: RequestValuationReport
  - Complex Type: RequestedPositions
  - Complex Type: TradeValuationItem
  - Complex Type: ValuationReport
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ◦ fpml-msg-4-8.xsd
	  ◦ fpml-valuation-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-valuation-4-8.xsd"/>
```

```
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
```

```
            </simpleType>
          </element>
        </sequence>
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

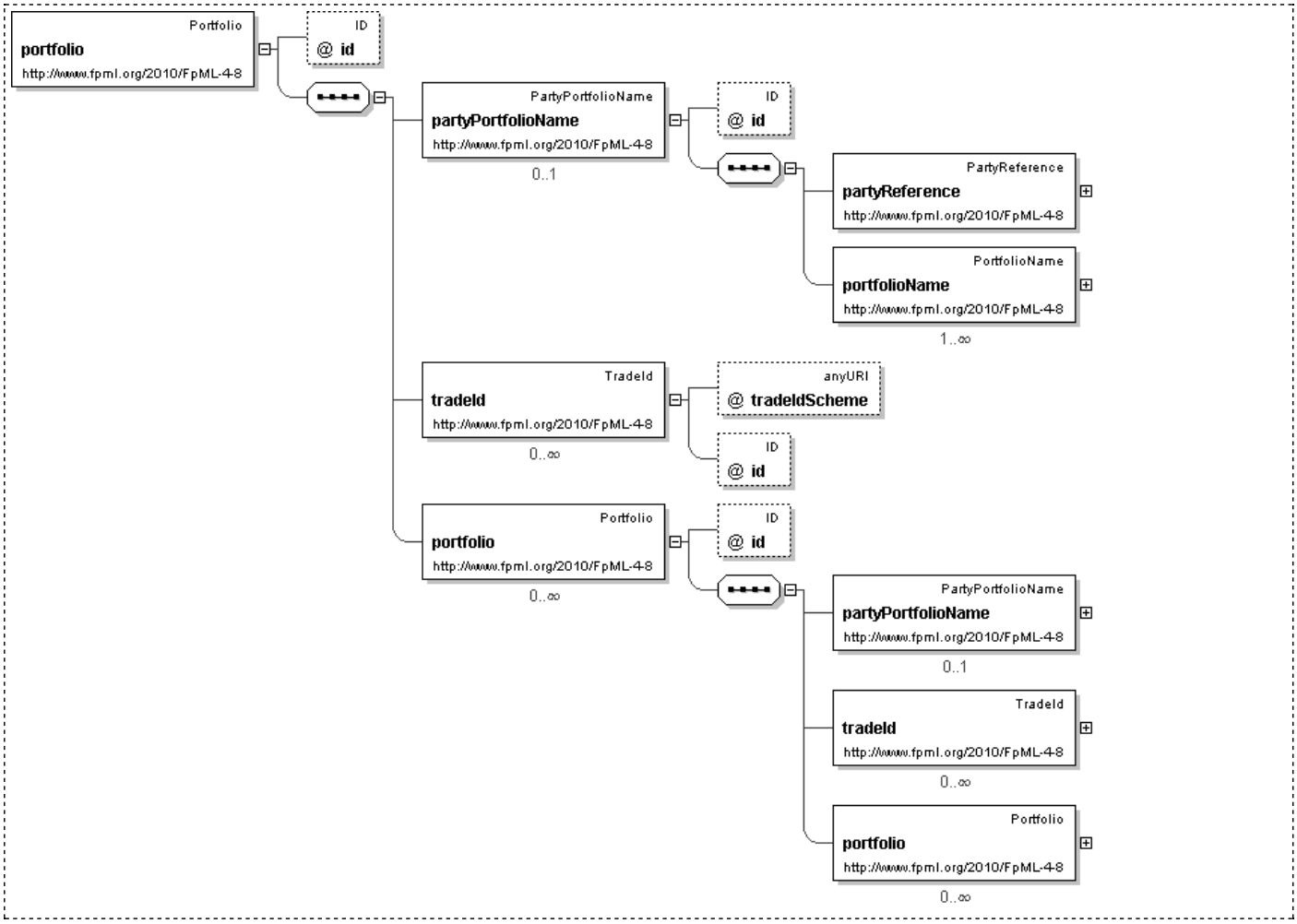
Element: portfolio

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - queryPortfolio

Name	portfolio
Used by (from the same schema document)	Complex Type PortfolioValuationItem
Type	Portfolio
Nillable	no
Abstract	no
Documentation	Global portfolio element used as a basis for a substitution group.

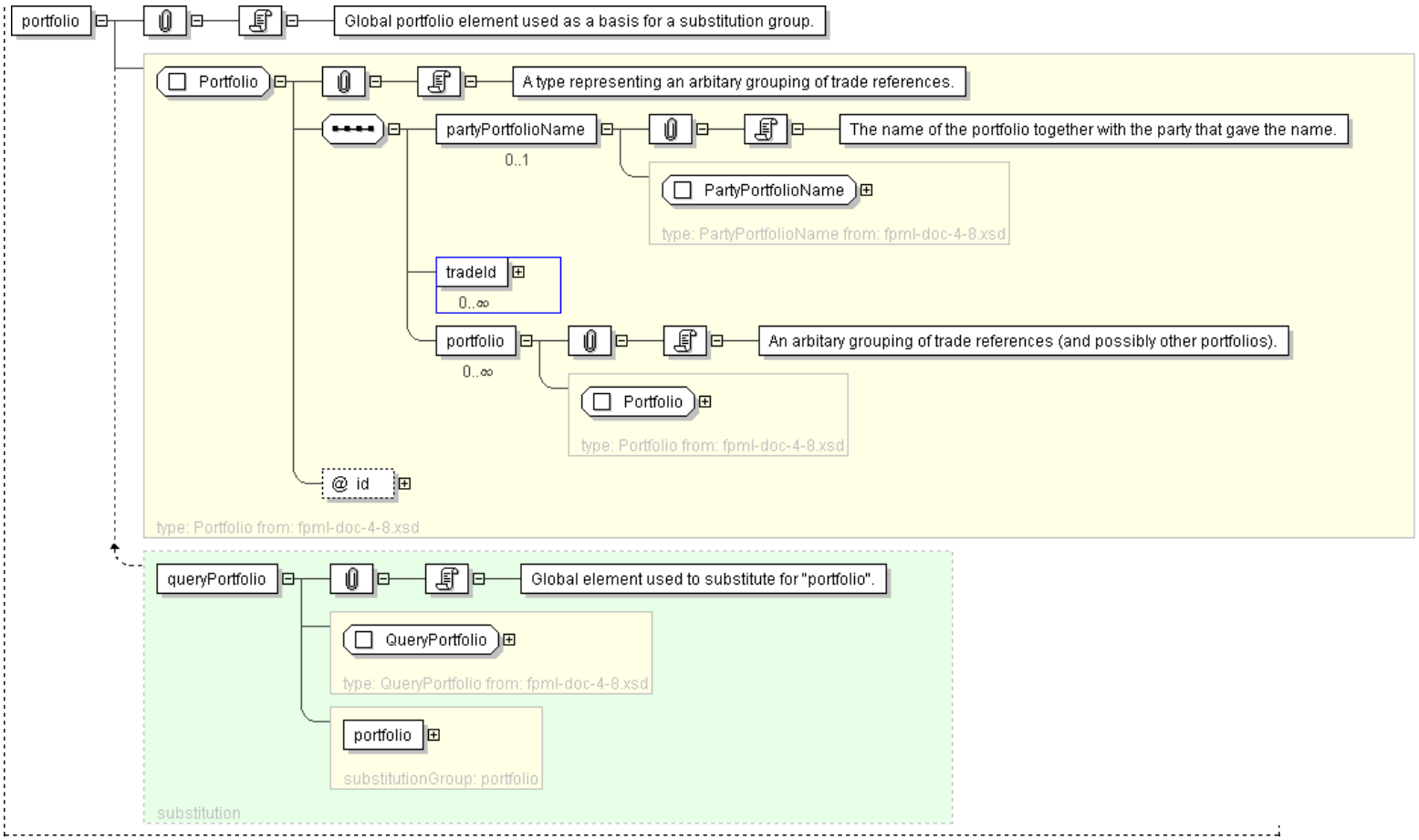
Logical Diagram



XML Instance Representation

```
<portfolio
 id=" xsd:ID [0..1]">
   <partyPortfolioName> PartyPortfolioName </partyPortfolioName> [0..1]
   'The name of the portfolio together with the party that gave the name.'

   <tradeId> TradeId </tradeId> [0..*]
   <portfolio> Portfolio </portfolio> [0..*]
   'An arbitary grouping of trade references (and possibly other portfolios).'

</portfolio>
```

Diagram



Schema Component Representation

```
<xsd:element name="portfolio" type=" Portfolio "/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

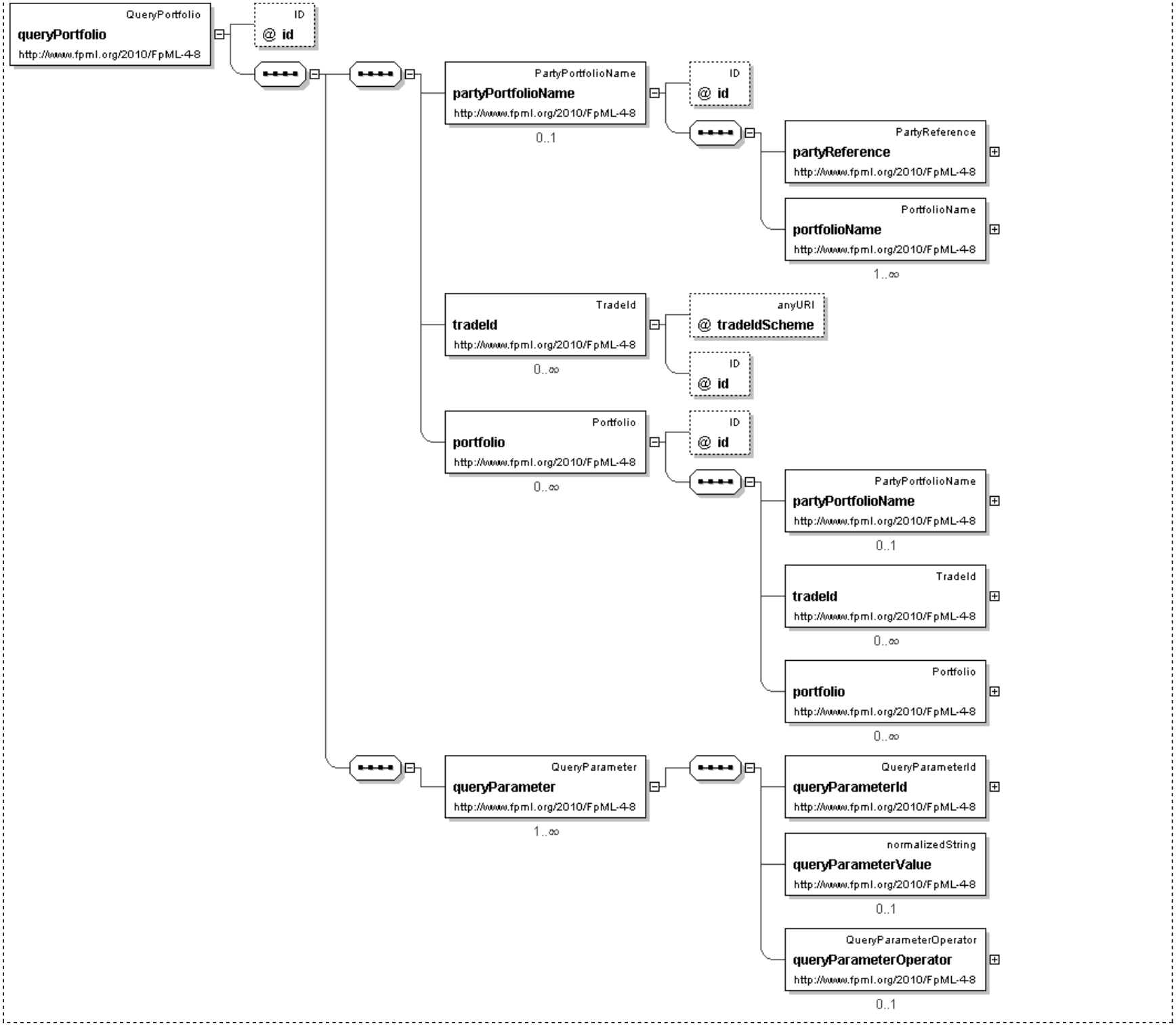
Element: queryPortfolio

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - portfolio

Name	queryPortfolio
Type	QueryPortfolio
Nillable	no
Abstract	no
Documentation	Global element used to substitute for "portfolio".

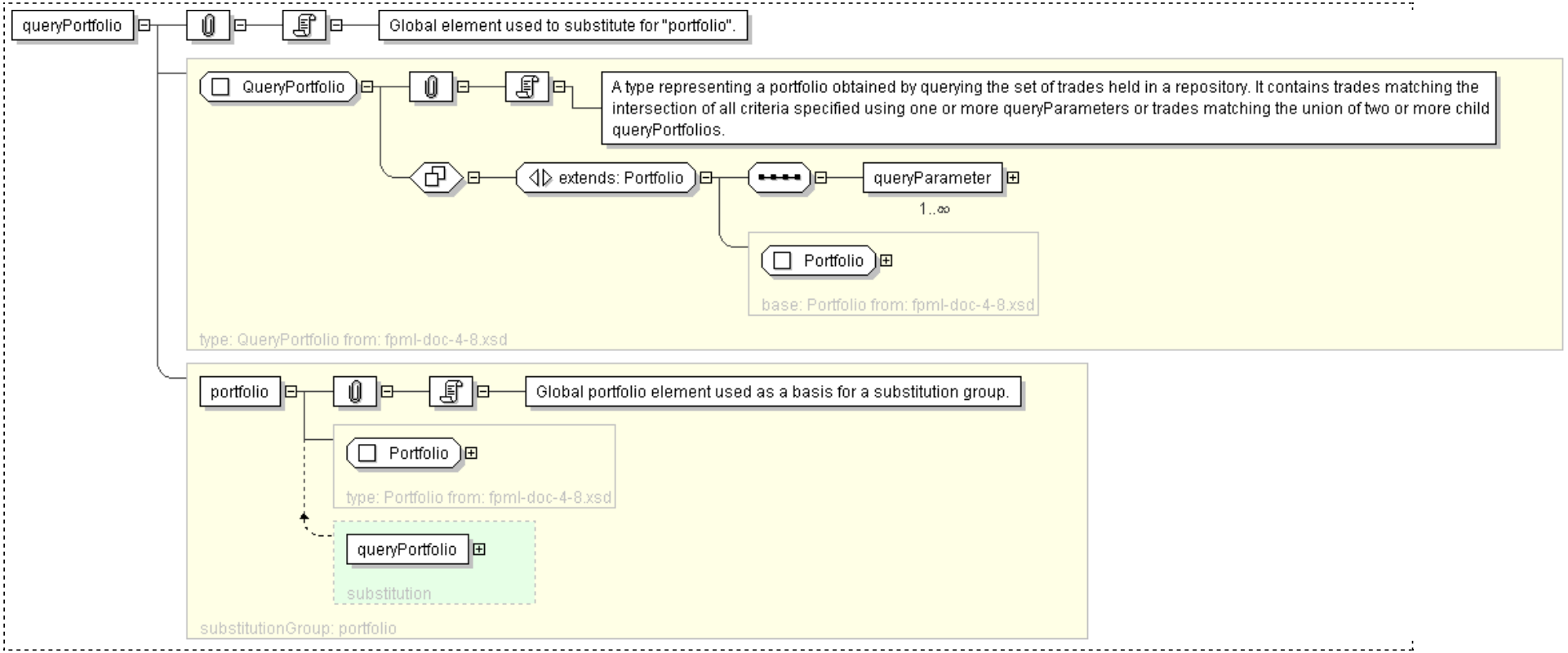
Logical Diagram


QueryPortfolio
queryPortfolio
http://www.fpml.org/2010/FpML-4-8
ID
@ id
PartyPortfolioName
partyPortfolioName
http://www.fpml.org/2010/FpML-4-8
0..1
ID
@ id
PartyReference
partyReference
http://www.fpml.org/2010/FpML-4-8
PortfolioName
portfolioName
http://www.fpml.org/2010/FpML-4-8
1..∞
TradeId
tradeId
http://www.fpml.org/2010/FpML-4-8
0..∞
anyURI
@ tradeIdScheme
ID
@ id
Portfolio
portfolio
http://www.fpml.org/2010/FpML-4-8
0..∞
ID
@ id
PartyPortfolioName
partyPortfolioName
http://www.fpml.org/2010/FpML-4-8
0..1
TradeId
tradeId
http://www.fpml.org/2010/FpML-4-8
0..∞
Portfolio
portfolio
http://www.fpml.org/2010/FpML-4-8
0..∞
QueryParameter
queryParameter
http://www.fpml.org/2010/FpML-4-8
1..∞
QueryParameterId
queryParameterId
http://www.fpml.org/2010/FpML-4-8
normalizedString
queryParameterValue
http://www.fpml.org/2010/FpML-4-8
0..1
QueryParameterOperator
queryParameterOperator
http://www.fpml.org/2010/FpML-4-8
0..1

XML Instance Representation

```
<queryPortfolio
 id=" xsd:ID [0..1]">
   <partyPortfolioName> PartyPortfolioName </partyPortfolioName> [0..1]
   'The name of the portfolio together with the party that gave the name.'

   <tradeId> TradeId </tradeId> [0..*]
   <portfolio> Portfolio </portfolio> [0..*]
   'An arbitary grouping of trade references (and possibly other portfolios).'

   <queryParameter> QueryParameter </queryParameter> [1..*]
</queryPortfolio>
```

Diagram



Schema Component Representation

```
<xsd:element name="queryPortfolio" type=" QueryPortfolio " substitutionGroup="portfolio"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PortfolioValuationItem

[Table of contents]

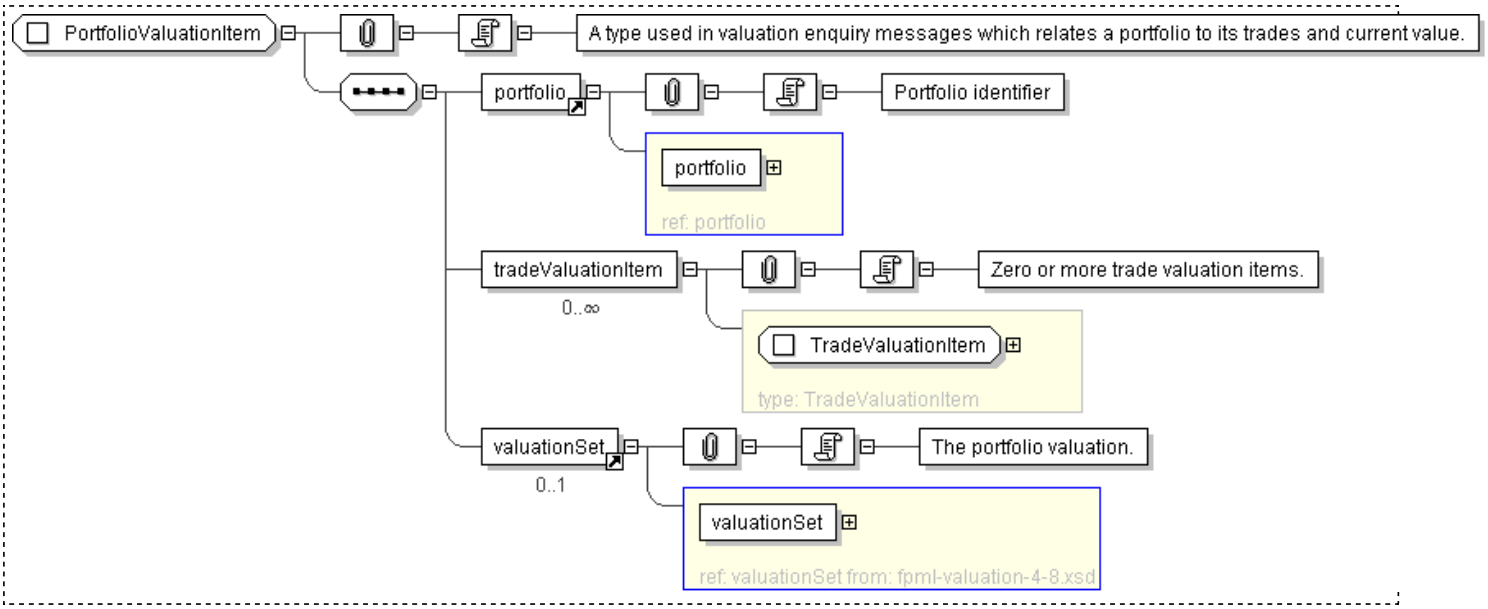
Super-types:	None
Sub-types:	None

Name	PortfolioValuationItem
Used by (from the same schema document)	Complex Type RequestValuationReport , Complex Type ValuationReport
Abstract	no
Documentation	A type used in valuation enquiry messages which relates a portfolio to its trades and current value.

XML Instance Representation

```
<...>
   <portfolio> ... </portfolio> [1]
   'Portfolio identifier'

   <tradeValuationItem> TradeValuationItem </tradeValuationItem> [0..*]
   'Zero or more trade valuation items.'

   <valuationSet> ... </valuationSet> [0..1]
   'The portfolio valuation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PortfolioValuationItem">
   <xsd:sequence>
      <xsd:element ref=" portfolio "/>
      <xsd:element name="tradeValuationItem" type=" TradeValuationItem " minOccurs="0"
      maxOccurs="unbounded"/>
      <xsd:element ref=" valuationSet " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionReport

[Table of contents]

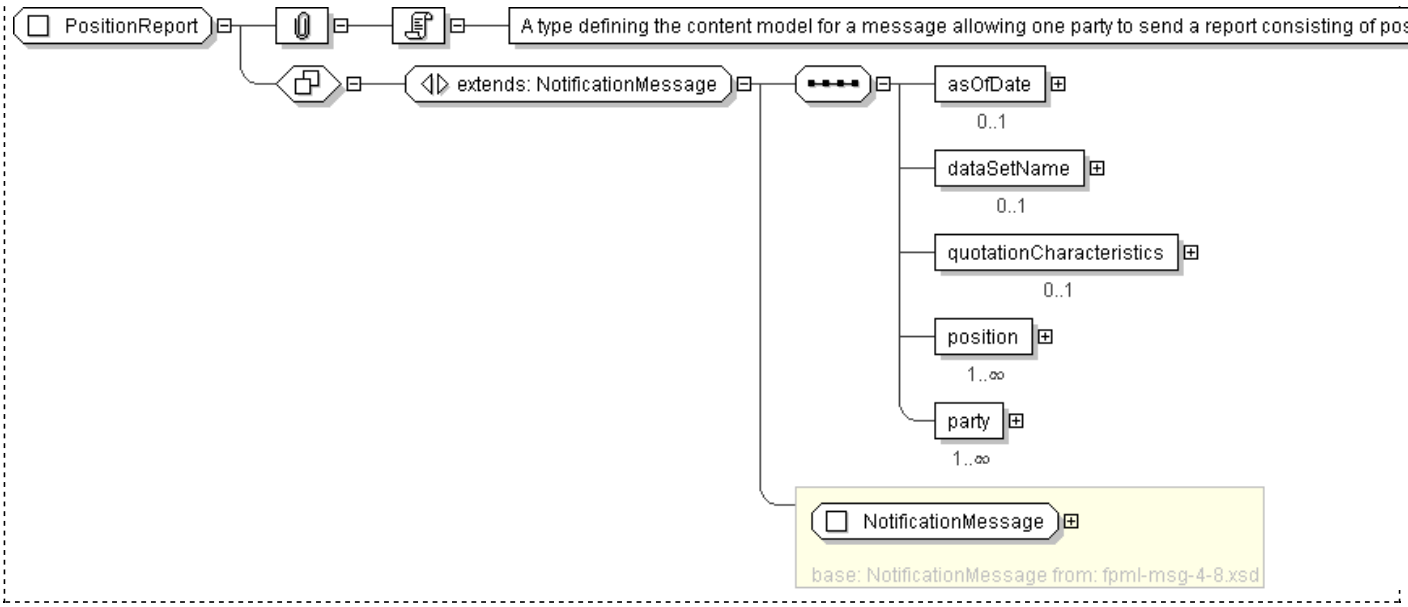
Super-types:	NotificationMessage < PositionReport (by extension)
Sub-types:	None

Name	PositionReport
Abstract	no
Documentation	A type defining the content model for a message allowing one party to send a report consisting of positions.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which
 build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance
 document. Instead, it is supplied by the XML parser when the document is validated against the FpML
 schema and indicates the build number of the schema file. Every time FpML publishes a change to the
 schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial
 Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <asOfDate> IdentifiedDate </asOfDate> [0..1]
   'The date for which this document reports positions and valuations.'

   <dataSetName> xsd:string </dataSetName> [0..1]
   'The name of the data set (portfolio, product type, etc.) that this report corresponds to. Used to
   help document the contents of the report.'

   <quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..1]
   'The default quotation characteristics for this document (e.g. currency, location). Currency must be
   specified; other fields may be specified.'

   <position> Position </position> [1..*]
   'The positions included in the position report.'

   <party> Party </party> [1..*]
   'The parties whose trades are included included in this position report.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionReport">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="asOfDate" type=" IdentifiedDate " minOccurs="0"/>
        <xsd:element name="dataSetName" type=" xsd:string " minOccurs="0"/>
        <xsd:element name="quotationCharacteristics" type=" QuotationCharacteristics " minOccurs="0"/>
        <xsd:element name="position" type=" Position " maxOccurs="unbounded"/>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestedPositions

[Table of contents]

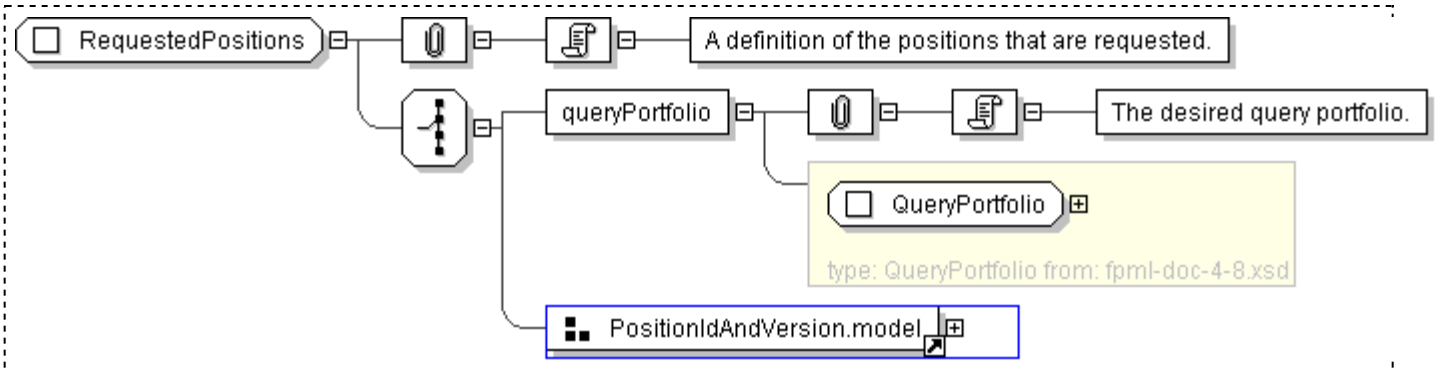
Super-types:	None
Sub-types:	None

Name	RequestedPositions
Used by (from the same schema document)	Complex Type RequestPositionReport
Abstract	no
Documentation	A definition of the positions that are requested.

XML Instance Representation

```
<...>
  Start Choice [1]
    <queryPortfolio> QueryPortfolio </queryPortfolio> [1]
     'The desired query portfolio.'

    <positionId> PositionId </positionId> [1]
     'A version-independent identifier for the position, possibly based on trade
     identifier.'

    <version> xsd:positiveInteger </version> [0..1]
     'A version identifier. Version identifiers must be ascending, i.e. higher
     numbers imply newer versions. There is no requirement that version
     identifiers for a position be sequential or small, so for example timestamp-
     based version identifiers could be used.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestedPositions">
  <xsd:choice>
    <xsd:element name="queryPortfolio" type=" QueryPortfolio "/>
    <xsd:group ref=" PositionIdAndVersion.model "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestPositionReport

[Table of contents]

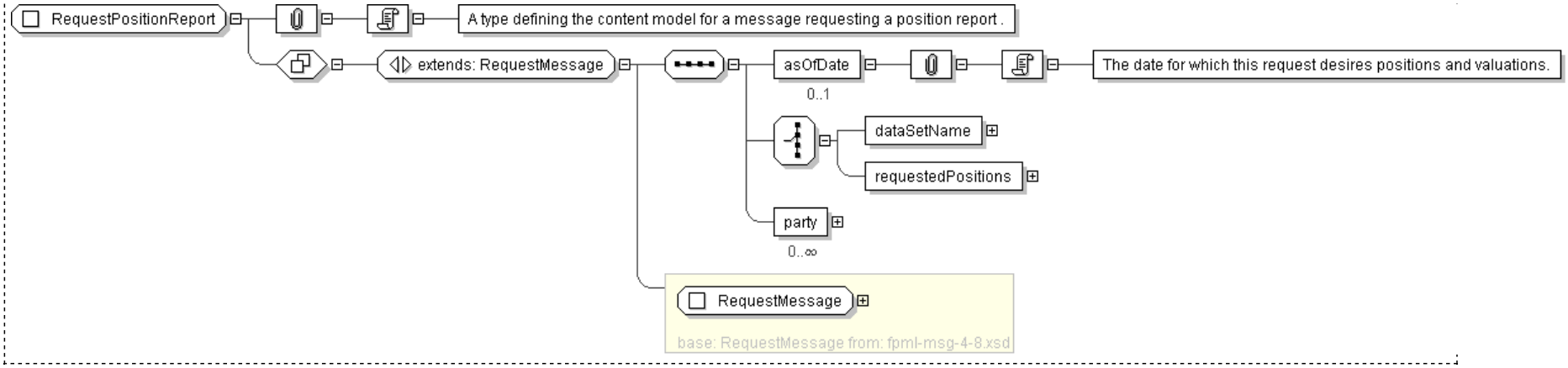
Super-types:	RequestMessage < RequestPositionReport (by extension)
Sub-types:	None

Name	RequestPositionReport
Abstract	no
Documentation	A type defining the content model for a message requesting a position report .

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to
 define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the
 XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML
 publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is
 incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual
 build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <asOfDate> ... </asOfDate> [0..1]
   'The date for which this request desires positions and valuations.'

   Start Choice [1]
      <dataSetName> xsd:normalizedString </dataSetName> [1]
      'The name of the data set (portfolio, product type, etc.) that this request corresponds to. Describes the desired report.'

      <requestedPositions> RequestedPositions </requestedPositions> [1]
      'The name of the data set (portfolio, product type, etc.) that this request corresponds to. Describes the desired report.'

   End Choice
   <party> Party </party> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestPositionReport">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="asOfDate" minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="dataSetName" type=" xsd:normalizedString "/>
          <xsd:element name="requestedPositions" type=" RequestedPositions "/>
        </xsd:choice>
        <xsd:element name="party" type=" Party " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestValuationReport

[Table of contents]

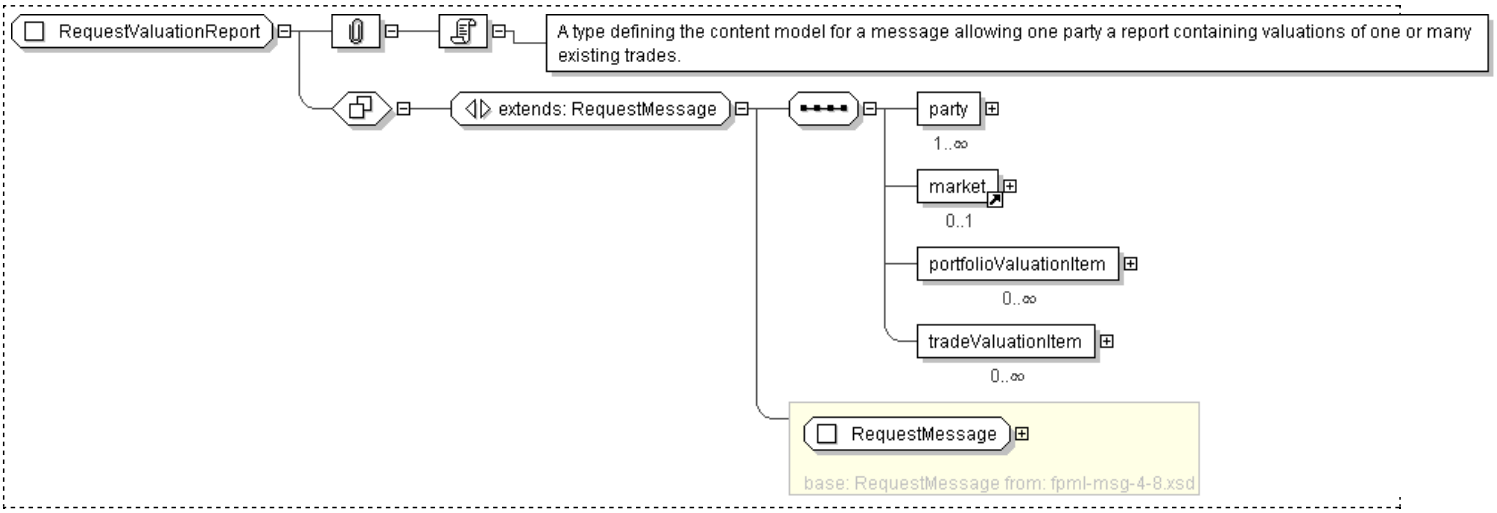
Super-types:	RequestMessage < RequestValuationReport (by extension)
Sub-types:	None

Name	RequestValuationReport
Abstract	no
Documentation	A type defining the content model for a message allowing one party a report containing valuations of one or many existing trades.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number
stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <party> Party </party> [1..*]
   <market> ... </market> [0..1]
   <portfolioValuationItem> PortfolioValuationItem </portfolioValuationItem> [0..*]
   'An instance of a unique portfolio valuation.'

   <tradeValuationItem> TradeValuationItem </tradeValuationItem> [0..*]
   'An instance of a unique trade valuation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestValuationReport">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
        <xsd:element ref=" market " minOccurs="0"/>
        <xsd:element name="portfolioValuationItem" type=" PortfolioValuationItem " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="tradeValuationItem" type=" TradeValuationItem " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeValuationItem

[Table of contents]

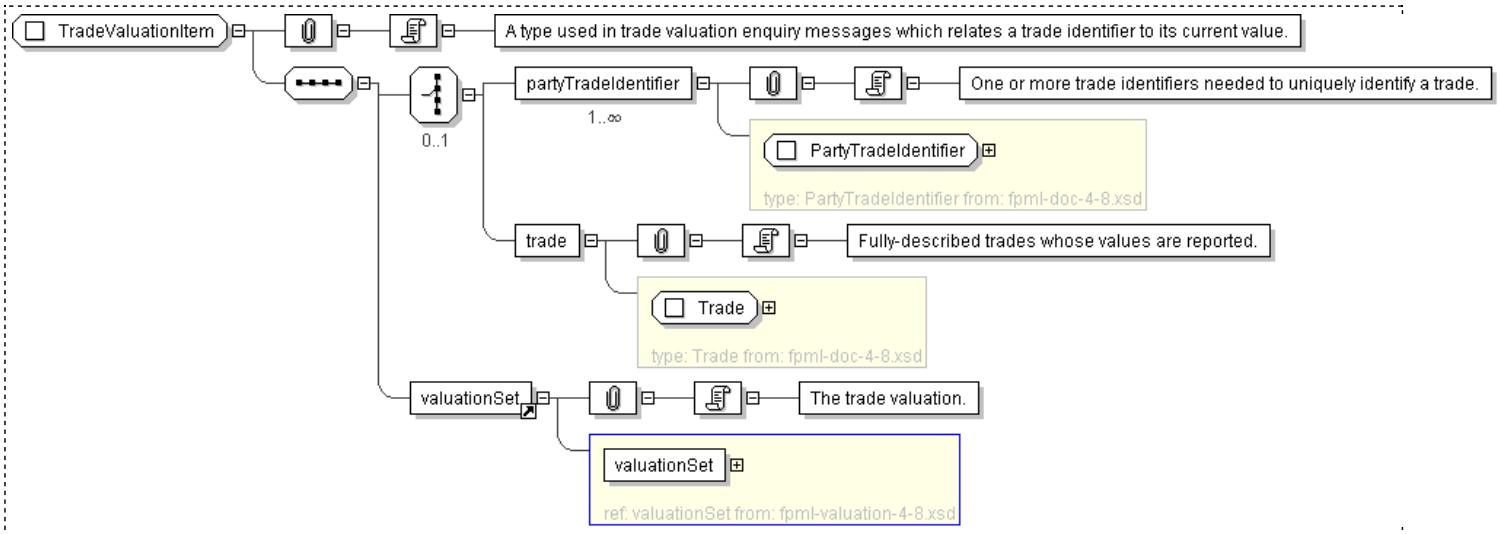
Super-types:	None
Sub-types:	None

Name	TradeValuationItem
Used by (from the same schema document)	Complex Type PortfolioValuationItem , Complex Type RequestValuationReport , Complex Type ValuationReport
Abstract	no
Documentation	A type used in trade valuation enquiry messages which relates a trade identifier to its current value.

XML Instance Representation

```
<...>
  Start Choice [0..1]
    <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1..*]
    'One or more trade identifiers needed to uniquely identify a trade.'

    <trade> Trade </trade> [1]
    'Fully-described trades whose values are reported.'

  End Choice
  <valuationSet> ... </valuationSet> [1]
  'The trade valuation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeValuationItem">
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier " maxOccurs="unbounded"/>
      <xsd:element name="trade" type=" Trade "/>
    </xsd:choice>
    <xsd:element ref=" valuationSet "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationReport

[Table of contents]

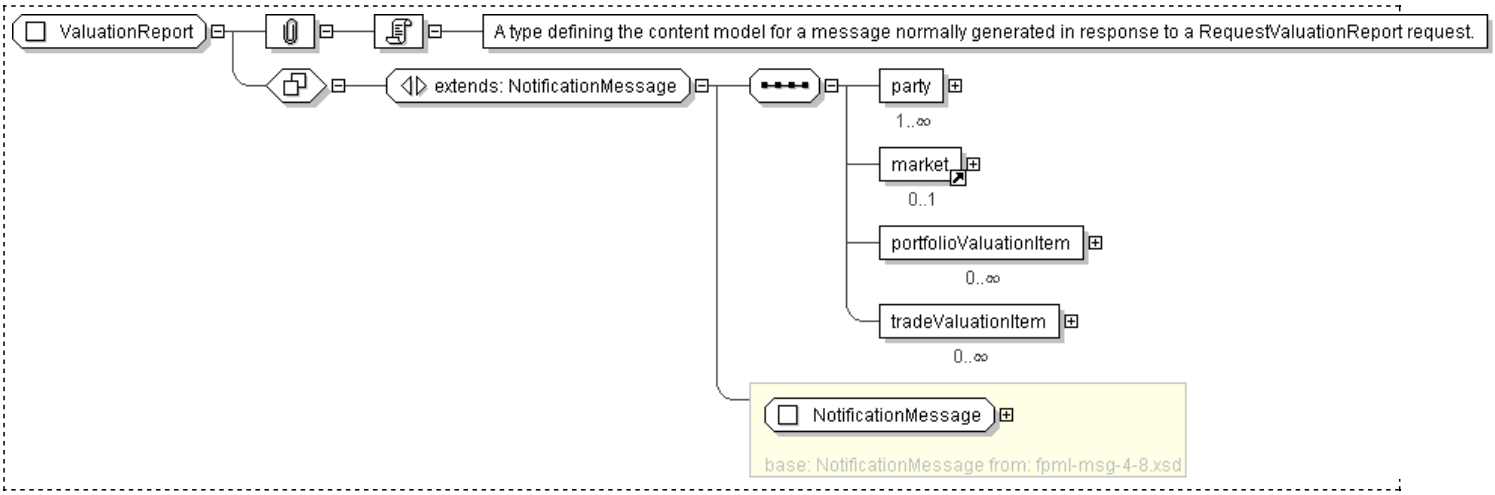
Super-types:	NotificationMessage < ValuationReport (by extension)
Sub-types:	None

Name	ValuationReport
Abstract	no
Documentation	A type defining the content model for a message normally generated in response to a RequestValuationReport request.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <party> Party </party> [1..*]
   <market> ... </market> [0..1]
   <portfolioValuationItem> PortfolioValuationItem </portfolioValuationItem> [0..*]
   'An instance of a unique portfolio valuation.'

   <tradeValuationItem> TradeValuationItem </tradeValuationItem> [0..*]
   'A collection of data values describing the state of the given trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationReport">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
        <xsd:element ref=" market " minOccurs="0"/>
        <xsd:element name="portfolioValuationItem" type=" PortfolioValuationItem " minOccurs="0"
        maxOccurs="unbounded"/>
        <xsd:element name="tradeValuationItem" type=" TradeValuationItem " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: equityLeg
  - Element: equitySwap
  - Element: equitySwapTransactionSupplement
- Global Definitions
  - Complex Type: DeprecatedEquityLeg
  - Complex Type: DeprecatedEquityLegValuation
  - Complex Type: DeprecatedEquityLegValuationPrice
  - Complex Type: DeprecatedEquityPaymentDates
  - Complex Type: EquitySwapTransactionSupplement
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ◦ fpml-ird-4-8.xsd
	  ◦ fpml-eq-shared-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-ird-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-eq-shared-4-8.xsd"/>
   ...
```

```
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
```

```
          </element>
        </sequence>
        <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

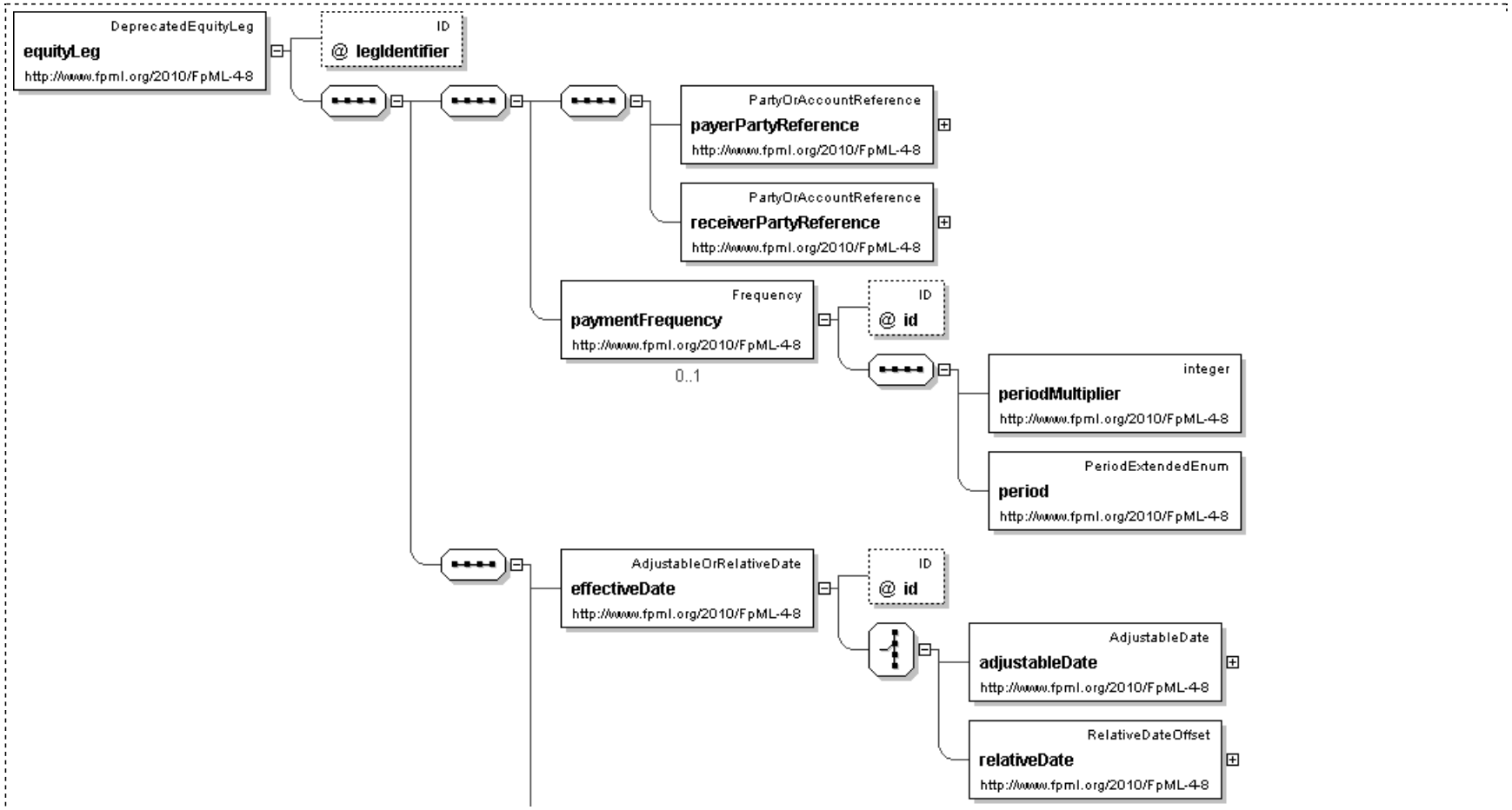
Element: equityLeg

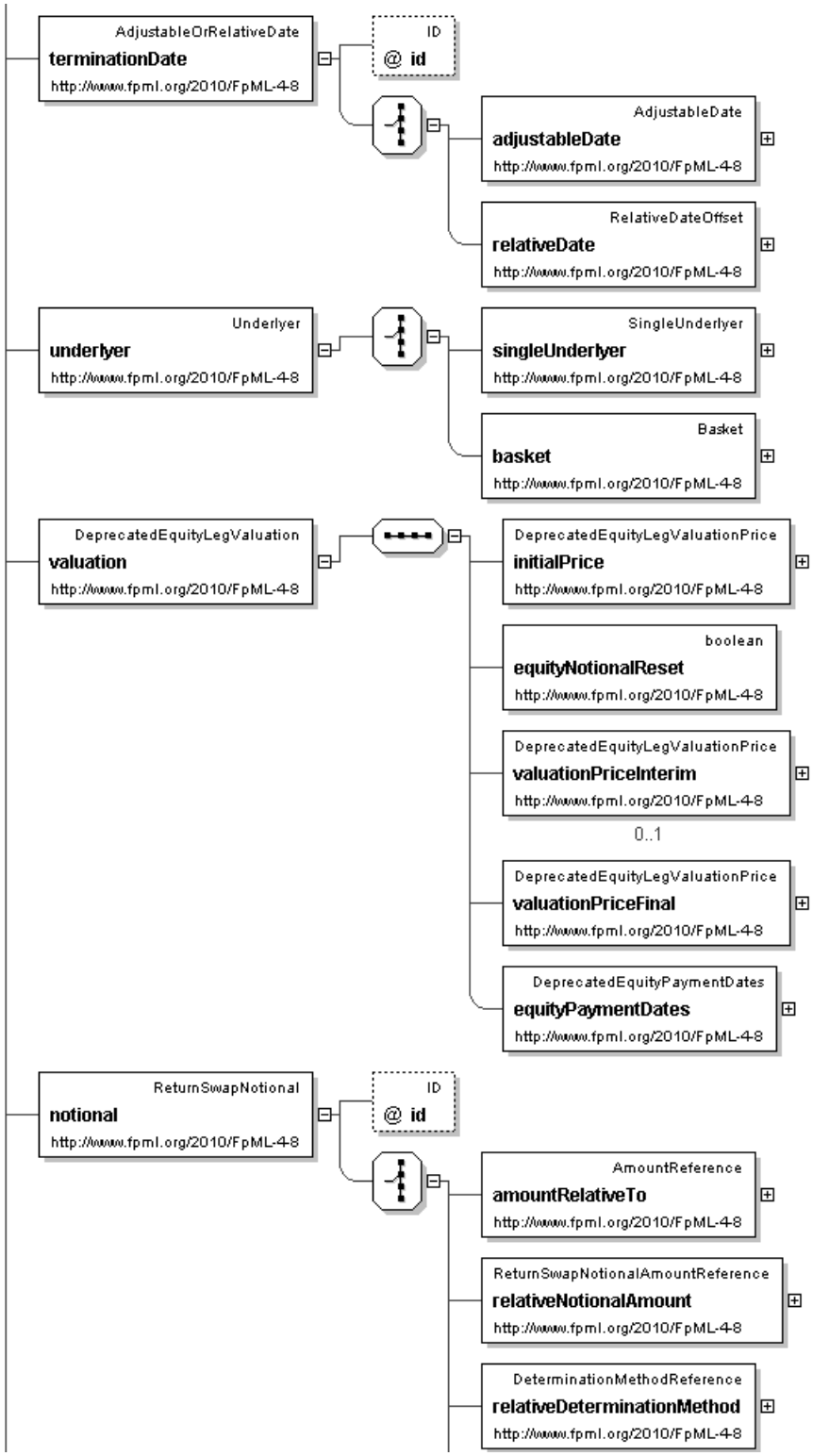
[Table of contents]

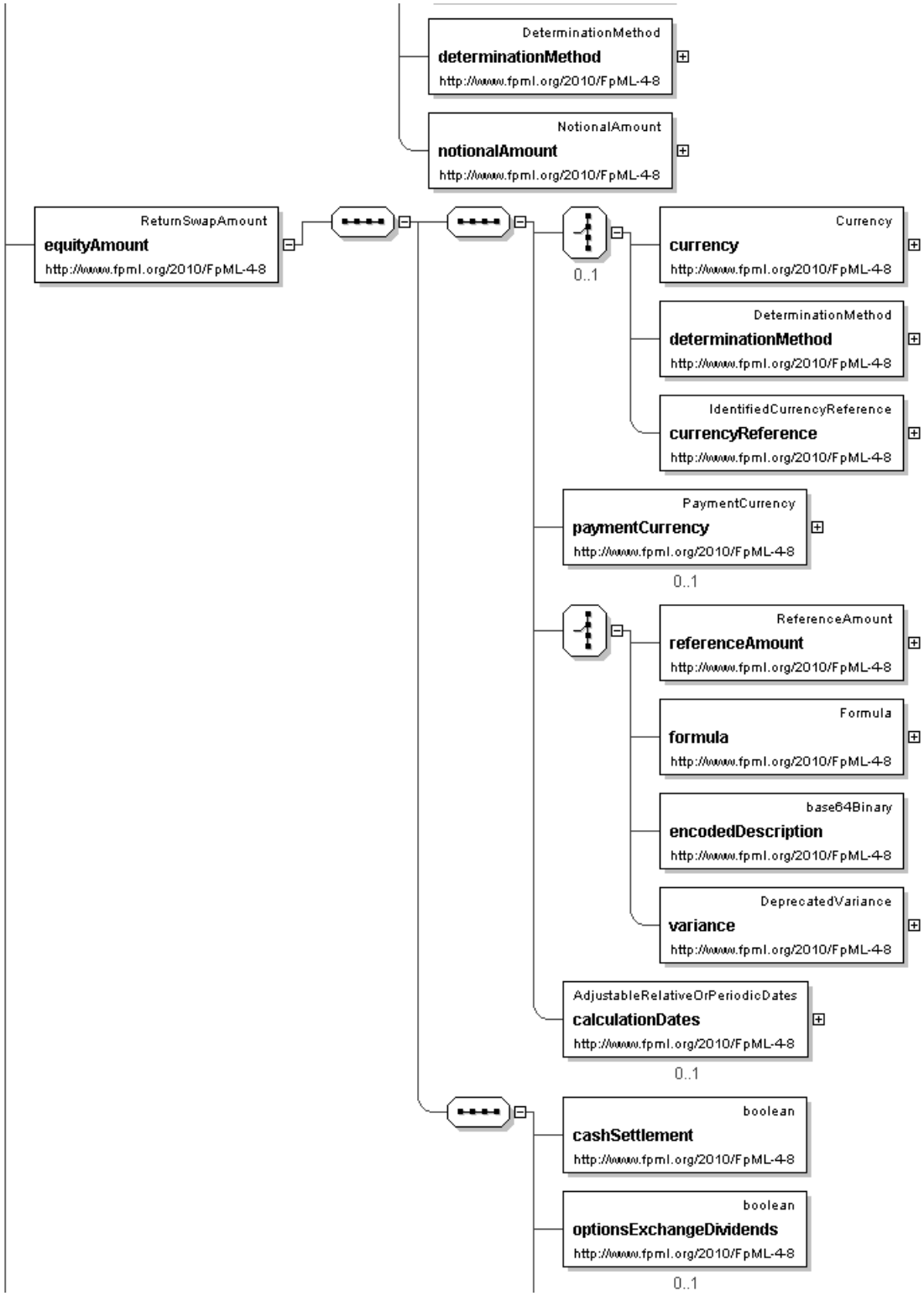
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

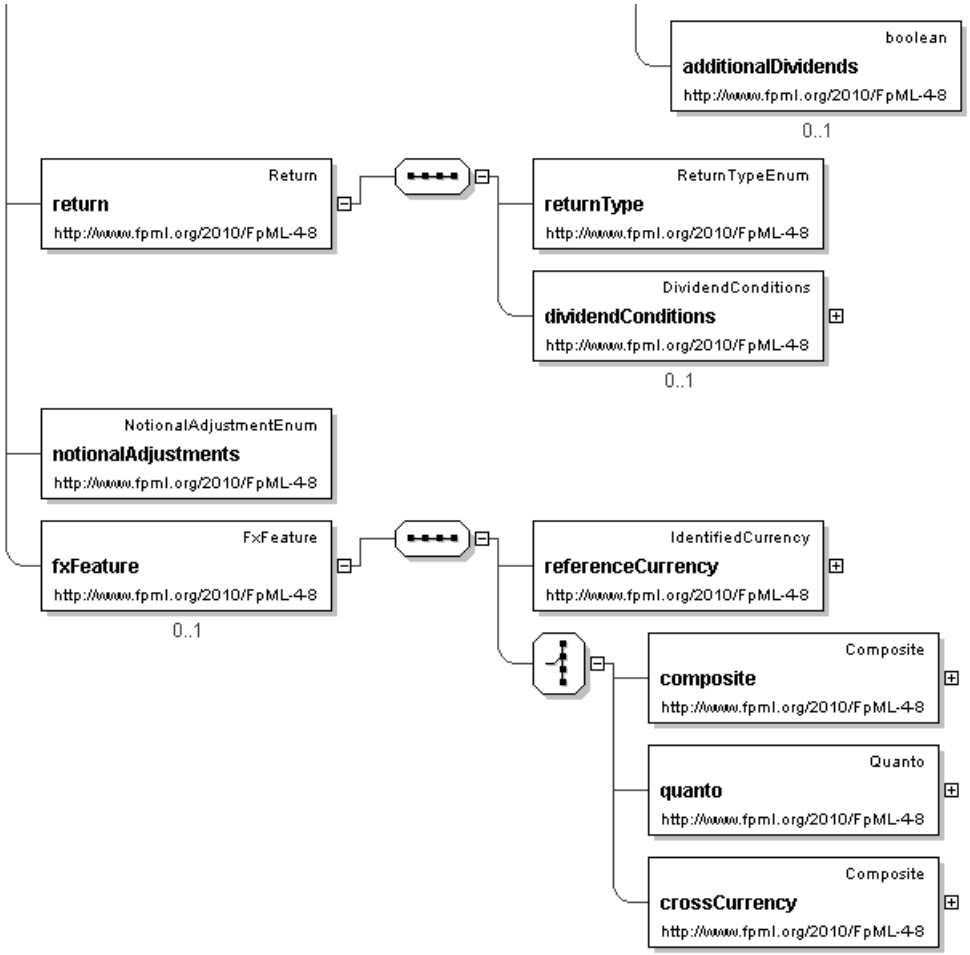
Name	equityLeg
Type	DeprecatedEquityLeg
Nillable	no
Abstract	no
Documentation	This element has been DEPRECATED and it will be removed in the next FpML major version (5.0) - please use returnLeg element to represent long form equity swaps, total return swaps. The equity amounts of the equity swap.

Logical Diagram




AdjustableOrRelativeDate
terminationDate
http://www.fpml.org/2010/FpML-4-8
ID
@ id
AdjustableDate
adjustableDate
http://www.fpml.org/2010/FpML-4-8
RelativeDateOffset
relativeDate
http://www.fpml.org/2010/FpML-4-8
Underlyer
underlyer
http://www.fpml.org/2010/FpML-4-8
SingleUnderlyer
singleUnderlyer
http://www.fpml.org/2010/FpML-4-8
Basket
basket
http://www.fpml.org/2010/FpML-4-8
DeprecatedEquityLegValuation
valuation
http://www.fpml.org/2010/FpML-4-8
DeprecatedEquityLegValuationPrice
initialPrice
http://www.fpml.org/2010/FpML-4-8
boolean
equityNotionalReset
http://www.fpml.org/2010/FpML-4-8
DeprecatedEquityLegValuationPrice
valuationPriceInterim
http://www.fpml.org/2010/FpML-4-8
0..1
DeprecatedEquityLegValuationPrice
valuationPriceFinal
http://www.fpml.org/2010/FpML-4-8
DeprecatedEquityPaymentDates
equityPaymentDates
http://www.fpml.org/2010/FpML-4-8
ReturnSwapNotional
notional
http://www.fpml.org/2010/FpML-4-8
ID
@ id
AmountReference
amountRelativeTo
http://www.fpml.org/2010/FpML-4-8
ReturnSwapNotionalAmountReference
relativeNotionalAmount
http://www.fpml.org/2010/FpML-4-8
DeterminationMethodReference
relativeDeterminationMethod
http://www.fpml.org/2010/FpML-4-8


DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
NotionalAmount
notionalAmount
http://www.fpml.org/2010/FpML-4-8
ReturnSwapAmount
equityAmount
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
DeterminationMethod
determinationMethod
http://www.fpml.org/2010/FpML-4-8
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2010/FpML-4-8
PaymentCurrency
paymentCurrency
http://www.fpml.org/2010/FpML-4-8
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2010/FpML-4-8
Formula
formula
http://www.fpml.org/2010/FpML-4-8
base64Binary
encodedDescription
http://www.fpml.org/2010/FpML-4-8
DeprecatedVariance
variance
http://www.fpml.org/2010/FpML-4-8
AdjustableRelativeOrPeriodicDates
calculationDates
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
cashSettlement
http://www.fpml.org/2010/FpML-4-8
boolean
optionsExchangeDividends
http://www.fpml.org/2010/FpML-4-8
0..1

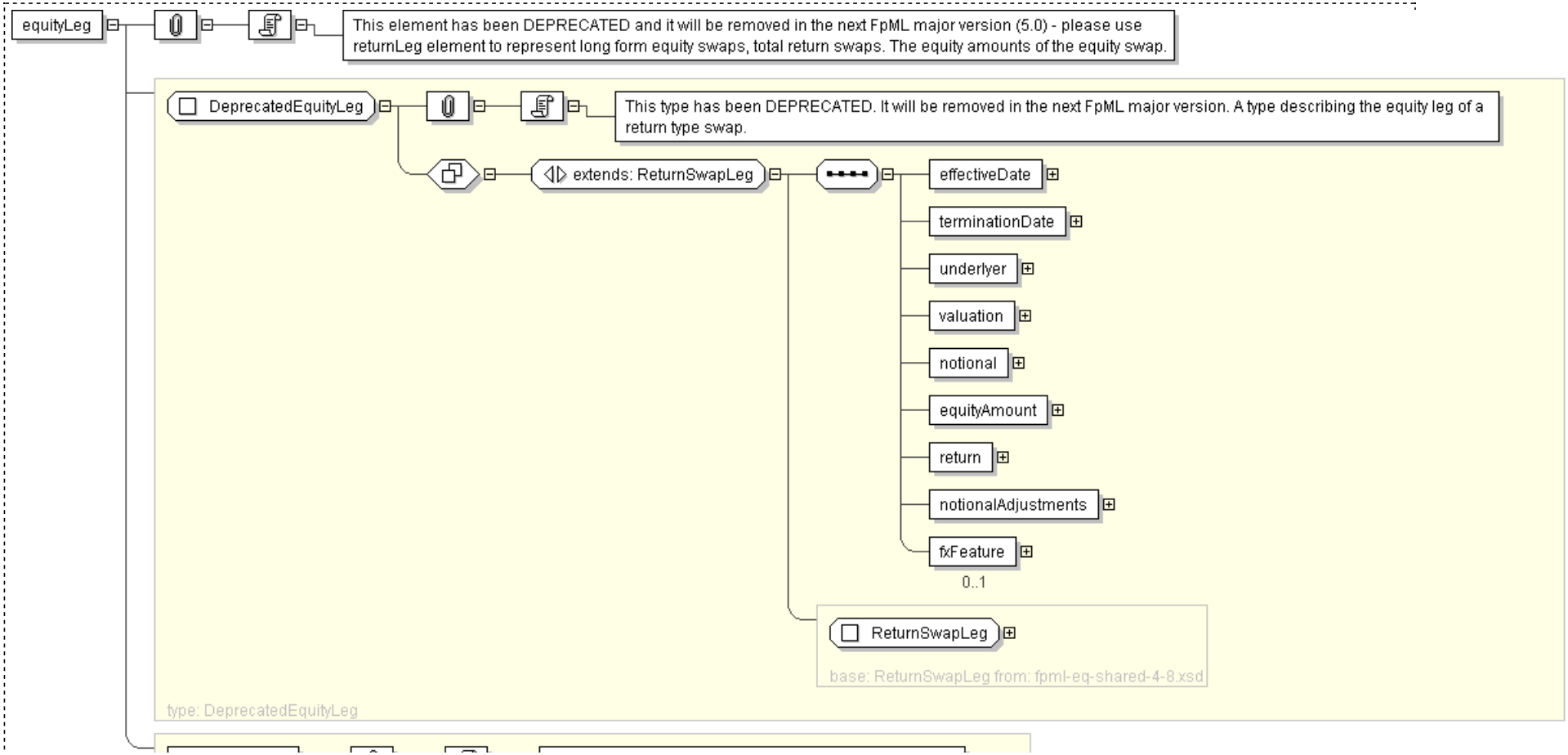


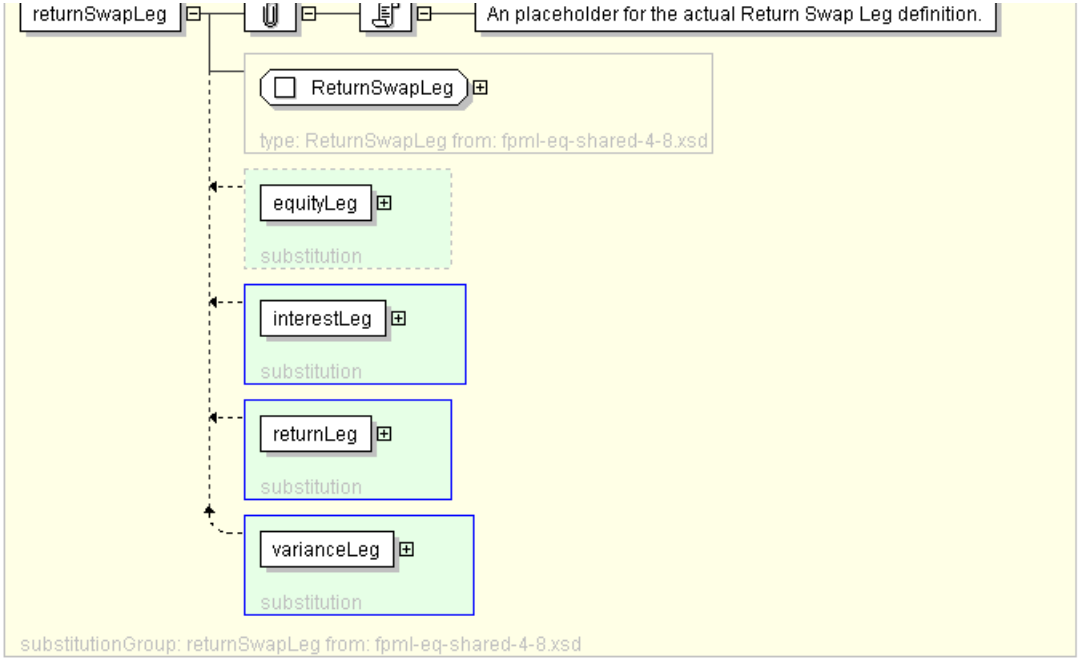
XML Instance Representation

```
<equityLeg
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of the equity leg of the swap. When defined in relation to a date specified somewhere else in the
   document (through the relativeDate component), this element will typically point to the effective date of the other leg of the
   swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of the equity leg of the swap. When defined in relation to a date specified somewhere else in
   the document (through the relativeDate component), this element will typically point to the termination date of the other leg of
   the swap.'
```

```
    <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlying component of the return type swap, which can be either one or many and consists in either equity,
    index or convertible bond component, or a combination of these.'

    <valuation> DeprecatedEquityLegValuation </valuation> [1]
    'Specifies the terms of the initial price of the return type swap and of the subsequent valuations of the equity underlyer.'

    <notional> ReturnSwapNotional </notional> [1]
    'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine the actual
    amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest
    leg, the definition will typically point to the definition of the equity leg.'

    <equityAmount> ReturnSwapAmount </equityAmount> [1]
    'Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates. Unless otherwise
    specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.'

    <return> Return </return> [1]
    'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the equity amounts.'

    <notionalAdjustments> NotionalAdjustmentEnum </notionalAdjustments> [1]
    'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

    <fxFeature> FxFeature </fxFeature> [0..1]
    'A quanto or composite FX feature.'

</equityLeg>
```

Diagram





Schema Component Representation

```
<xsd:element name="equityLeg" type=" DeprecatedEquityLeg " substitutionGroup="returnSwapLeg" deprecated="true" deprecatedReason="It has
been made more generic as returnLeg to cover return swap type products."/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

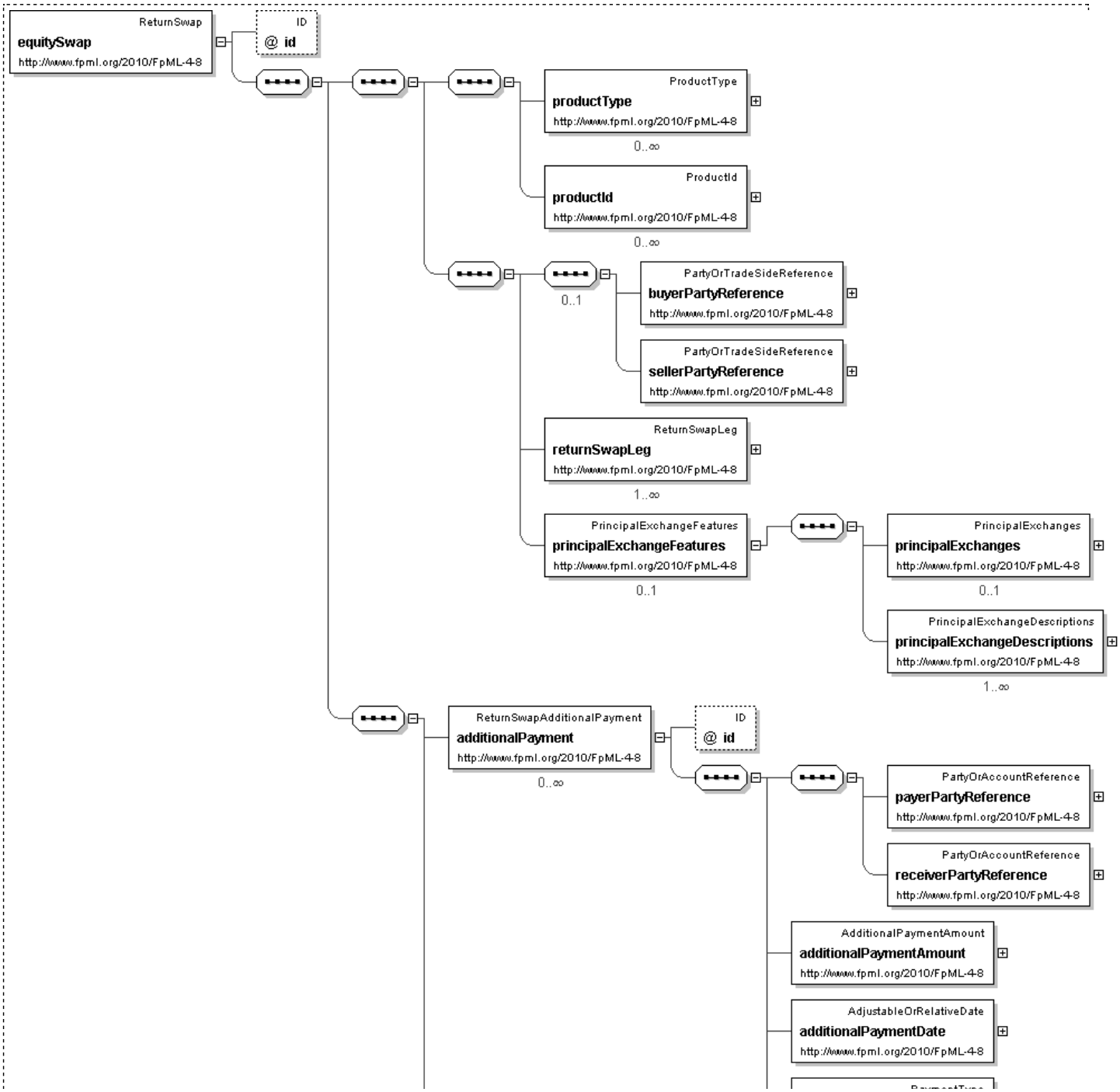
Element: equitySwap

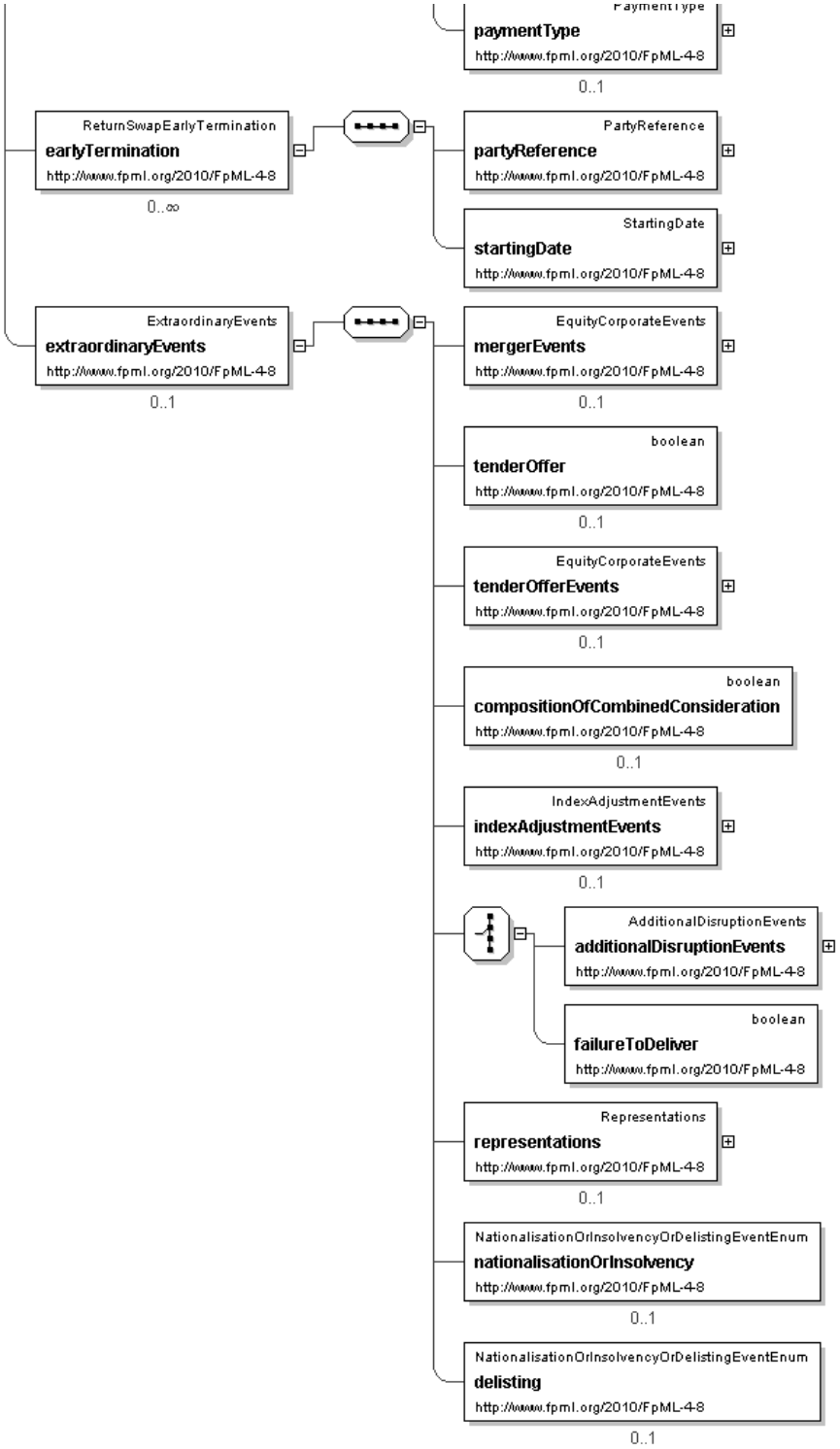
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	equitySwap
Type	ReturnSwap
Nillable	no
Abstract	no
Documentation	This element has been DEPRECATED and it will be removed in the next FpML major version (5.0) - please use returnSwap element to represent long form equity swaps, total return swaps, and variance swaps.

Logical Diagram





XML Instance Representation

```
<equitySwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this
   element. Note that the domain values for this element are not strictly an enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where an
implementor wishes to indicate who has manufactured the Swap through representing them as the Seller. It may be
removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights
      defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
```

'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

End Group: BuyerSeller.model
<returnSwapLeg> ... </returnSwapLeg> [1..*]
<principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
'This is used to document a Fully Funded Return Swap.'

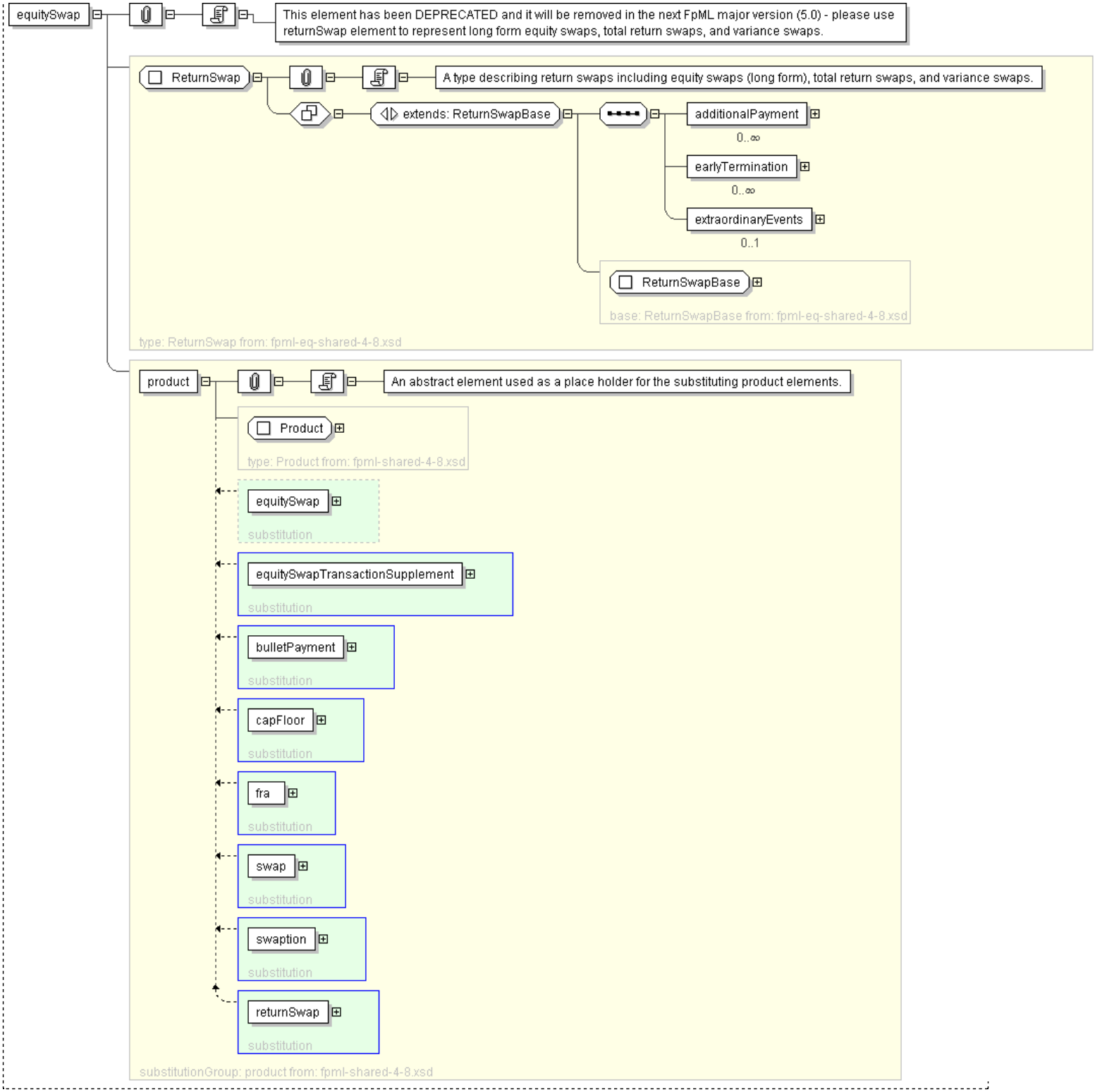
<additionalPayment> ReturnSwapAdditionalPayment </additionalPayment> [0..*]
'Specifies additional payment(s) between the principal parties to the trade. This component extends some of the features of the additionalPayment component developed by the FpML industry group. Appropriate discussions will determine whether it would be appropriate to extend the shared component in order to meet the further requirements of equity swaps.'

<earlyTermination> ReturnSwapEarlyTermination </earlyTermination> [0..*]
'Specifies, for one or for both the parties to the trade, the date from which it can early terminate it.'

<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.'

</equitySwap>

Diagram



Schema Component Representation

```
<xsd:element name="equitySwap" type=" ReturnSwap " substitutionGroup="product" deprecated="true" deprecatedReason="It has
been made more generic as returnSwap to cover return swap type products."/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

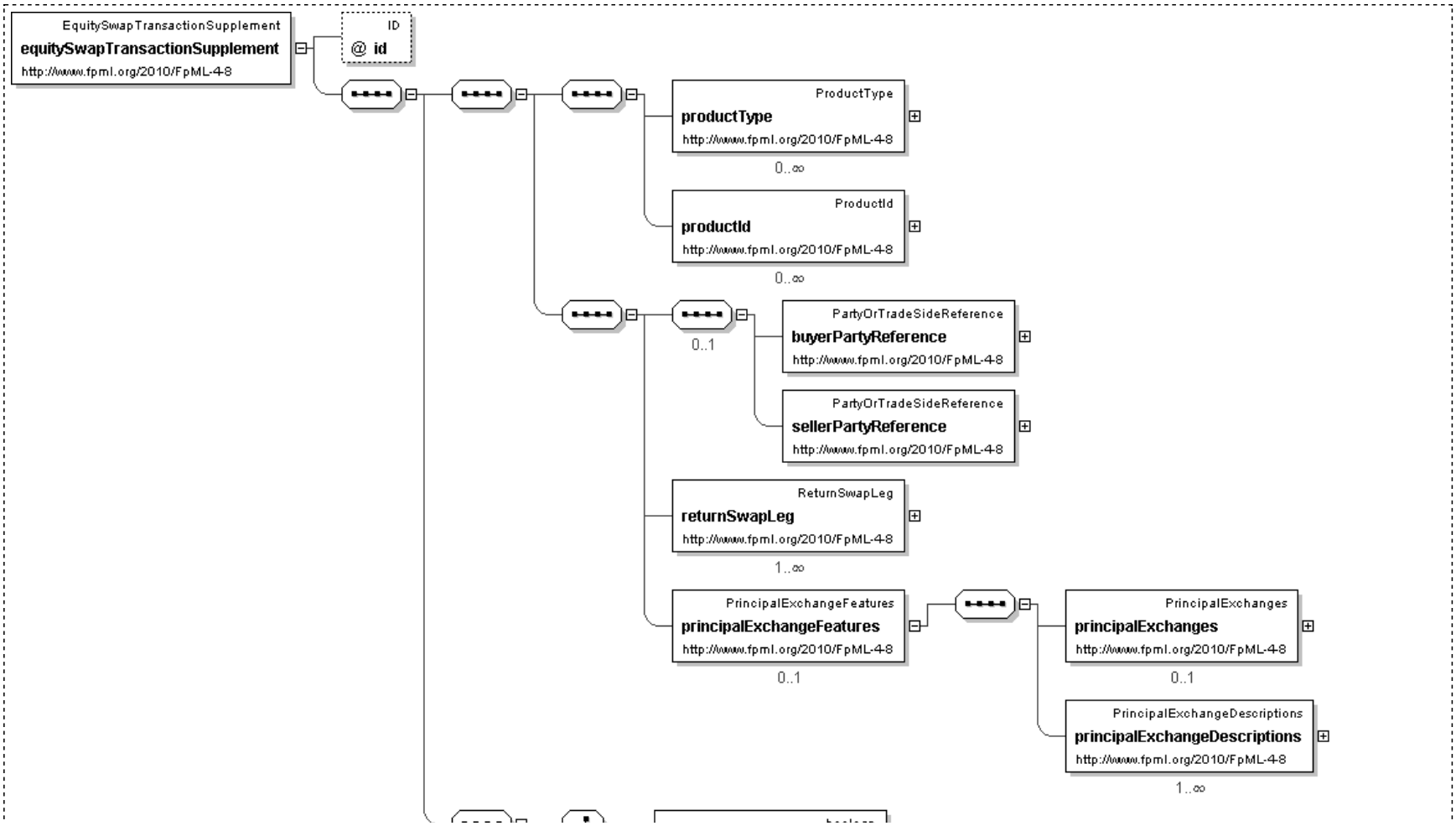
Element: equitySwapTransactionSupplement

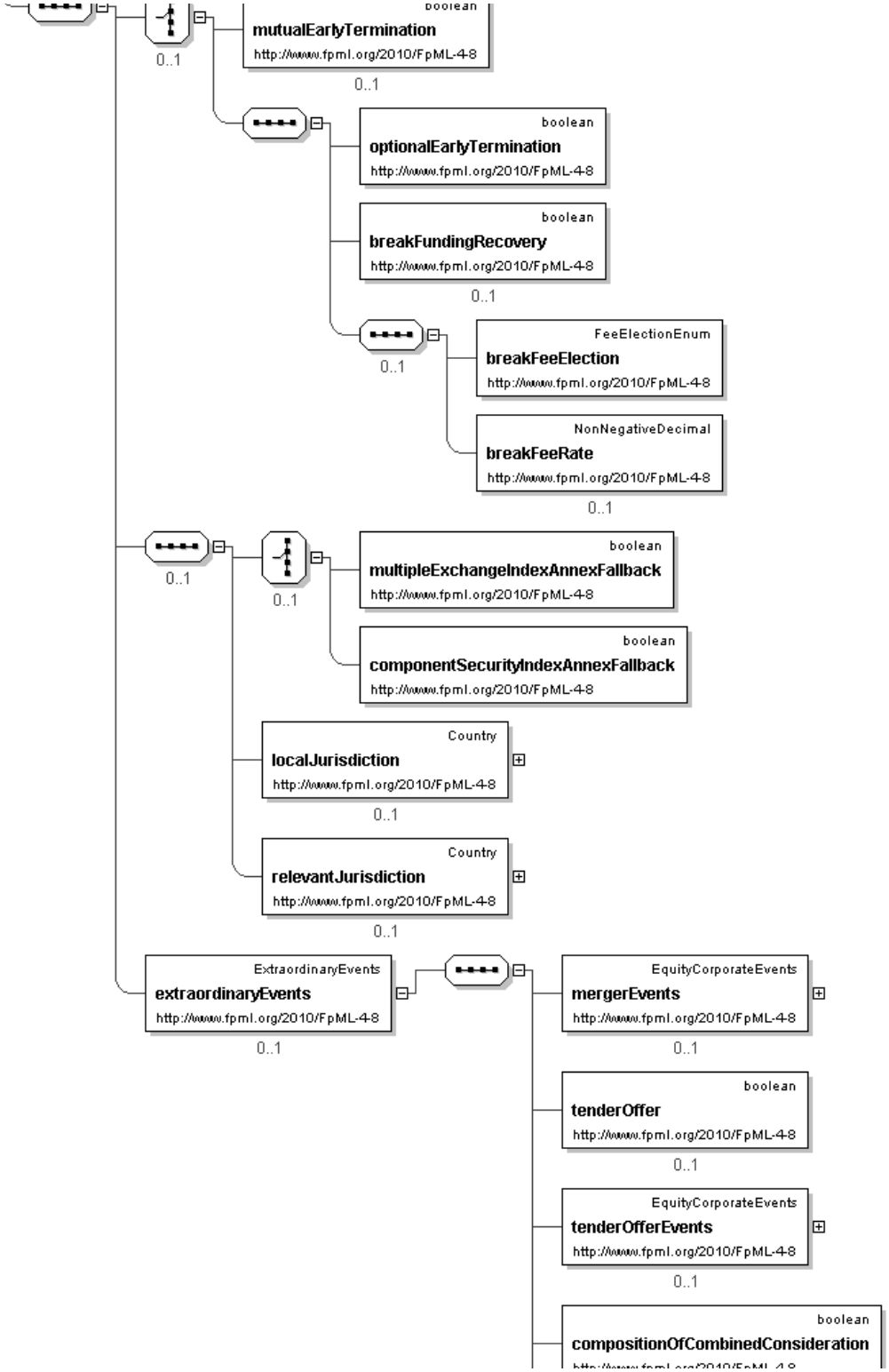
[Table of contents]

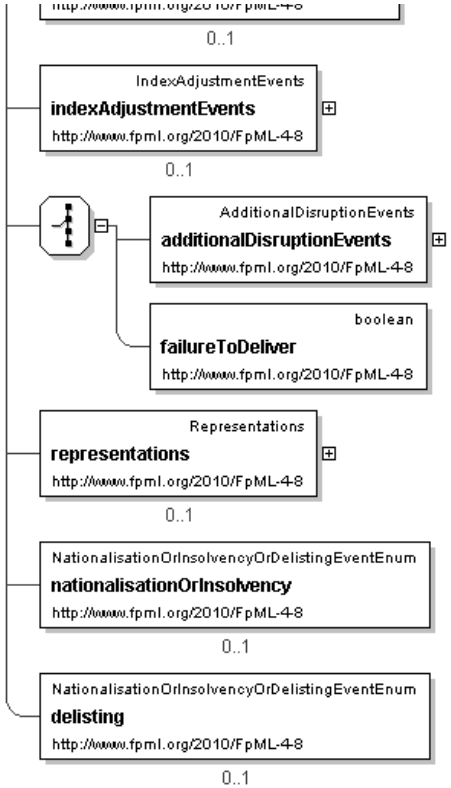
- *This element can be used wherever the following element is referenced:*
  - product

Name	equitySwapTransactionSupplement
Type	EquitySwapTransactionSupplement
Nillable	no
Abstract	no
Documentation	Specifies the structure of the equity swap transaction supplement.

Logical Diagram




boolean
mutualEarlyTermination
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
boolean
optionalEarlyTermination
http://www.fpml.org/2010/FpML-4-8
boolean
breakFundingRecovery
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
FeeElectionEnum
breakFeeElection
http://www.fpml.org/2010/FpML-4-8
NonNegativeDecimal
breakFeeRate
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
0..1
boolean
multipleExchangeIndexAnnexFallback
http://www.fpml.org/2010/FpML-4-8
boolean
componentSecurityIndexAnnexFallback
http://www.fpml.org/2010/FpML-4-8
Country
localJurisdiction
http://www.fpml.org/2010/FpML-4-8
0..1
Country
relevantJurisdiction
http://www.fpml.org/2010/FpML-4-8
0..1
ExtraordinaryEvents
extraordinaryEvents
http://www.fpml.org/2010/FpML-4-8
0..1
EquityCorporateEvents
mergerEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
tenderOffer
http://www.fpml.org/2010/FpML-4-8
0..1
EquityCorporateEvents
tenderOfferEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
compositionOfCombinedConsideration

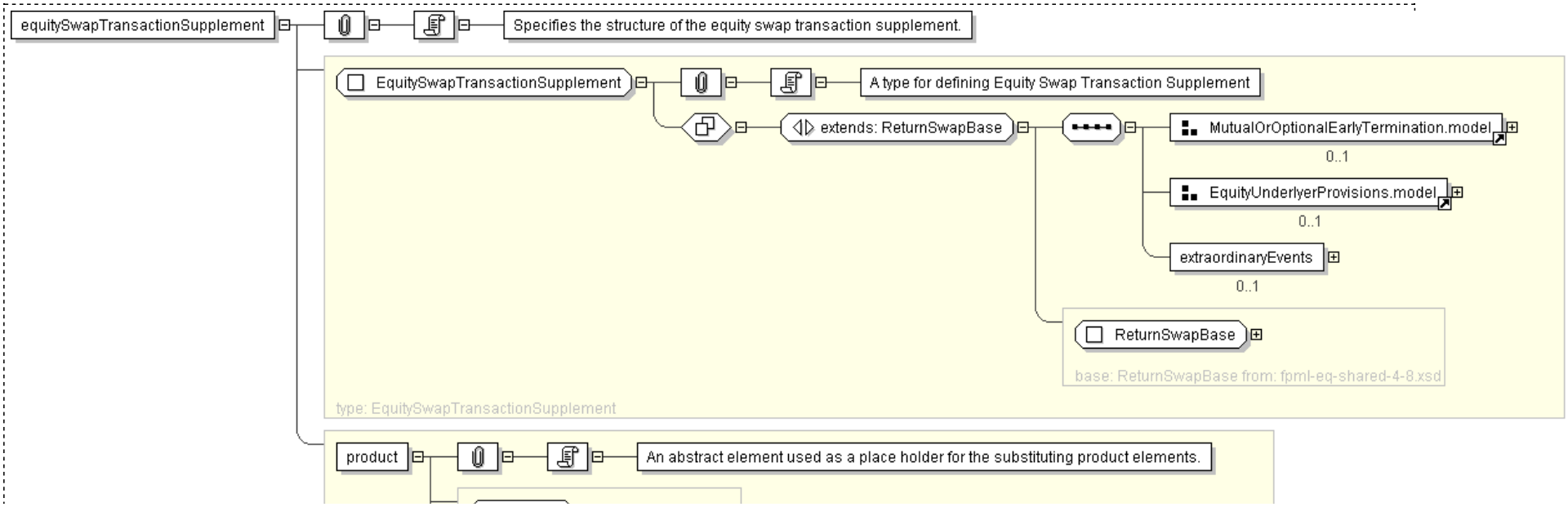


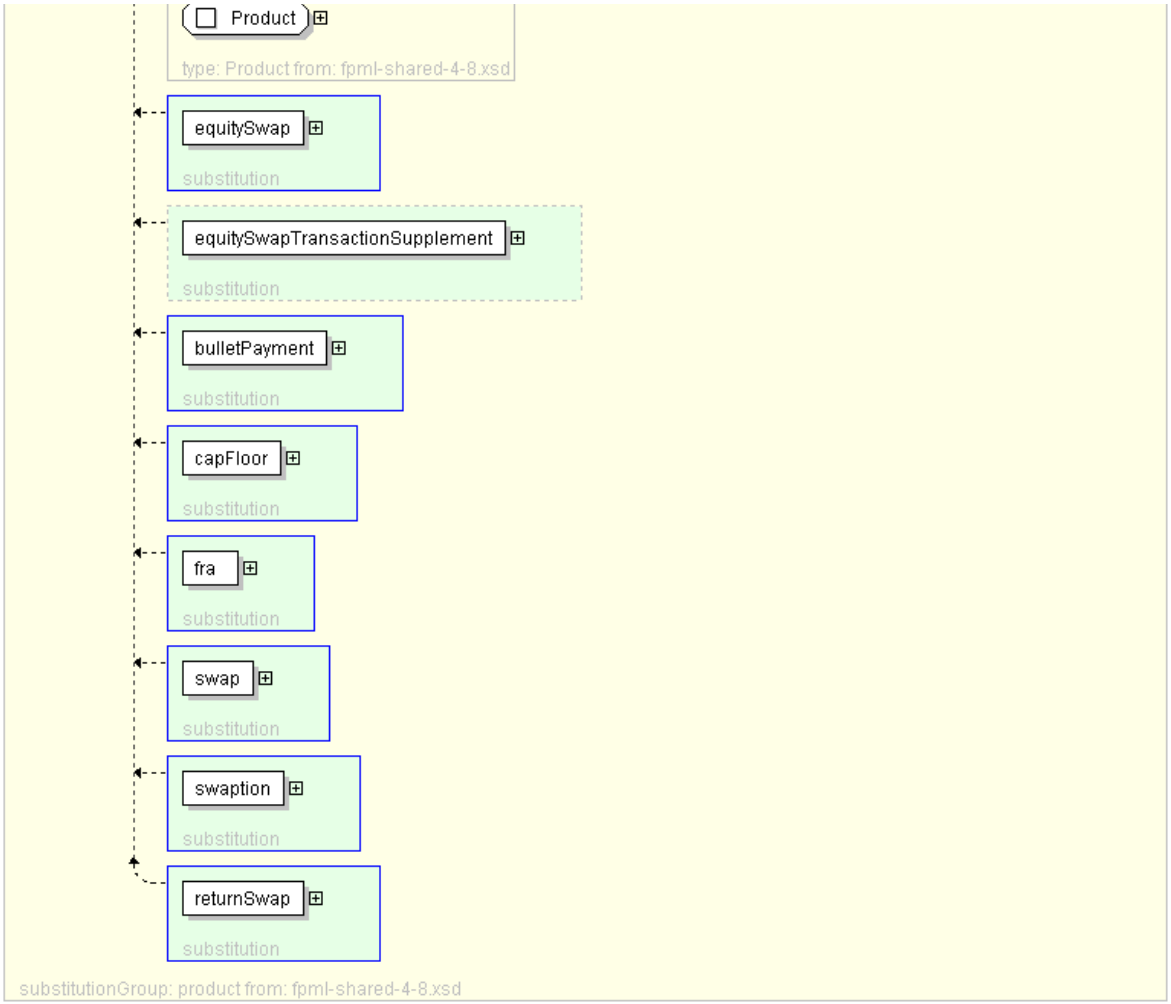
XML Instance Representation

```
<equitySwapTransactionSupplement
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that
   the domain values for this element are not strictly an enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where an implementor wishes to
indicate who has manufactured the Swap through representing them as the Seller. It may be removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000
      ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in
      return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   End Group: BuyerSeller.model
   <returnSwapLeg> ... </returnSwapLeg> [1..*]
   <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
   'This is used to document a Fully Funded Return Swap.'

   Start Group: MutualOrOptionalEarlyTermination.model [0..1]
      Start Choice [1]
         <mutualEarlyTermination> xsd:boolean </mutualEarlyTermination> [0..1]
         'Used for specifying whether the Mutual Early Termination Right that is detailed in the Master Confirmation will apply.'

         <optionalEarlyTermination> xsd:boolean </optionalEarlyTermination> [1]
```

```
        'A Boolean element used for specifying whether the Optional Early Termination clause detailed in the agreement will apply.'

        <breakFundingRecovery> xsd:boolean </breakFundingRecovery> [0..1]
        'A Boolean element used for specifying whether the Break Funding Recovery detailed in the agreement will apply.'

        Start Sequence [0..1]
           <breakFeeElection> FeeElectionEnum </breakFeeElection> [1]
           'Defines the fee type.'

           <breakFeeRate> NonNegativeDecimal </breakFeeRate> [0..1]
        End Sequence
     End Choice
  End Group: MutualOrOptionalEarlyTermination.model
  Start Group: EquityUnderlyerProvisions.model [0..1]
     Start Group: IndexAnnexFallback.model [0..1]
        Start Choice [1]
           <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
           'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to the
           transaction. This annex defines additional provisions which are applicable where an index is comprised of component securities
           that are traded on multiple exchanges.'

           <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
           'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to the
           transaction.'

        End Choice
     End Group: IndexAnnexFallback.model
     <localJurisdiction> Country </localJurisdiction> [0..1]
     'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean taxes,
     duties, and similar charges imposed by the taxing authority of the Local Jurisdiction If this element is not present Local
     Jurisdiction is Not Applicable.'

     <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
     'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean
     taxes, duties and similar charges that would be imposed by the taxing authority of the Country of Underlyer on a Hypothetical Broker
     Dealer assuming the Applicable Hedge Positions are held by its office in the Relevant Jurisdiction. If this element is not present
     Relevant Jurisdiction is Not Applicable.'

  End Group: EquityUnderlyerProvisions.model
  <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
  'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction
  to be adjusted.'

</equitySwapTransactionSupplement>
```

Diagram





Schema Component Representation

```
<xsd:element name="equitySwapTransactionSupplement" type=" EquitySwapTransactionSupplement " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedEquityLeg

[Table of contents]

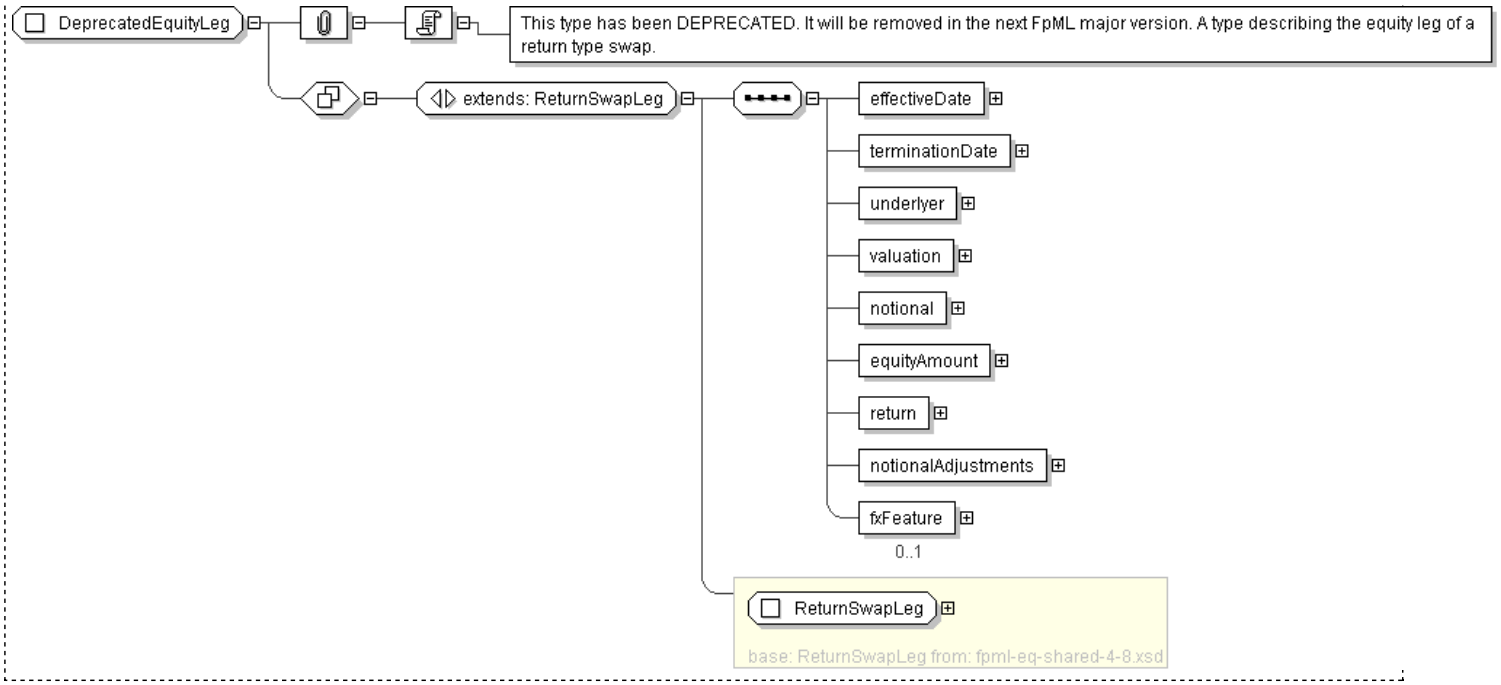
Super-types:	ReturnSwapLeg < DeprecatedEquityLeg (by extension)
Sub-types:	None

Name	DeprecatedEquityLeg
Used by (from the same schema document)	Element equityLeg
Abstract	no
Documentation	This type has been DEPRECATED. It will be removed in the next FpML major version. A type describing the equity leg of a return type swap.

XML Instance Representation

```
<...
 legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Frequency </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of the equity leg of the swap. When defined in relation to a date specified
   somewhere else in the document (through the relativeDate component), this element will typically point to the
   effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
   'Specifies the termination date of the equity leg of the swap. When defined in relation to a date specified
   somewhere else in the document (through the relativeDate component), this element will typically point to the
   termination date of the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlying component of the return type swap, which can be either one or many and consists in
   either equity, index or convertible bond component, or a combination of these.'

   <valuation> DeprecatedEquityLegValuation </valuation> [1]
   'Specifies the terms of the initial price of the return type swap and of the subsequent valuations of the equity
   underlyer.'

   <notional> ReturnSwapNotional </notional> [1]
   'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine
   the actual amount (using the notional component defined by the FpML industry group) and the determination method.
   When used in the interest leg, the definition will typically point to the definition of the equity leg.'

   <equityAmount> ReturnSwapAmount </equityAmount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates. Unless
   otherwise specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.'

   <return> Return </return> [1]
   'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the equity
   amounts.'

   <notionalAdjustments> NotionalAdjustmentEnum </notionalAdjustments> [1]
   'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

   <fxFeature> FxFeature </fxFeature> [0..1]
   'A quanto or composite FX feature.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedEquityLeg" deprecated="true" deprecatedReason="It has been made more generic as
ReturnLeg to cover return swap type products.">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="underlyer" type=" Underlyer "/>
            <xsd:element name="valuation" type=" DeprecatedEquityLegValuation "/>
            <xsd:element name="notional" type=" ReturnSwapNotional "/>
            <xsd:element name="equityAmount" type=" ReturnSwapAmount "/>
            <xsd:element name="return" type=" Return "/>
            <xsd:element name="notionalAdjustments" type=" NotionalAdjustmentEnum "/>
            <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedEquityLegValuation

[Table of contents]

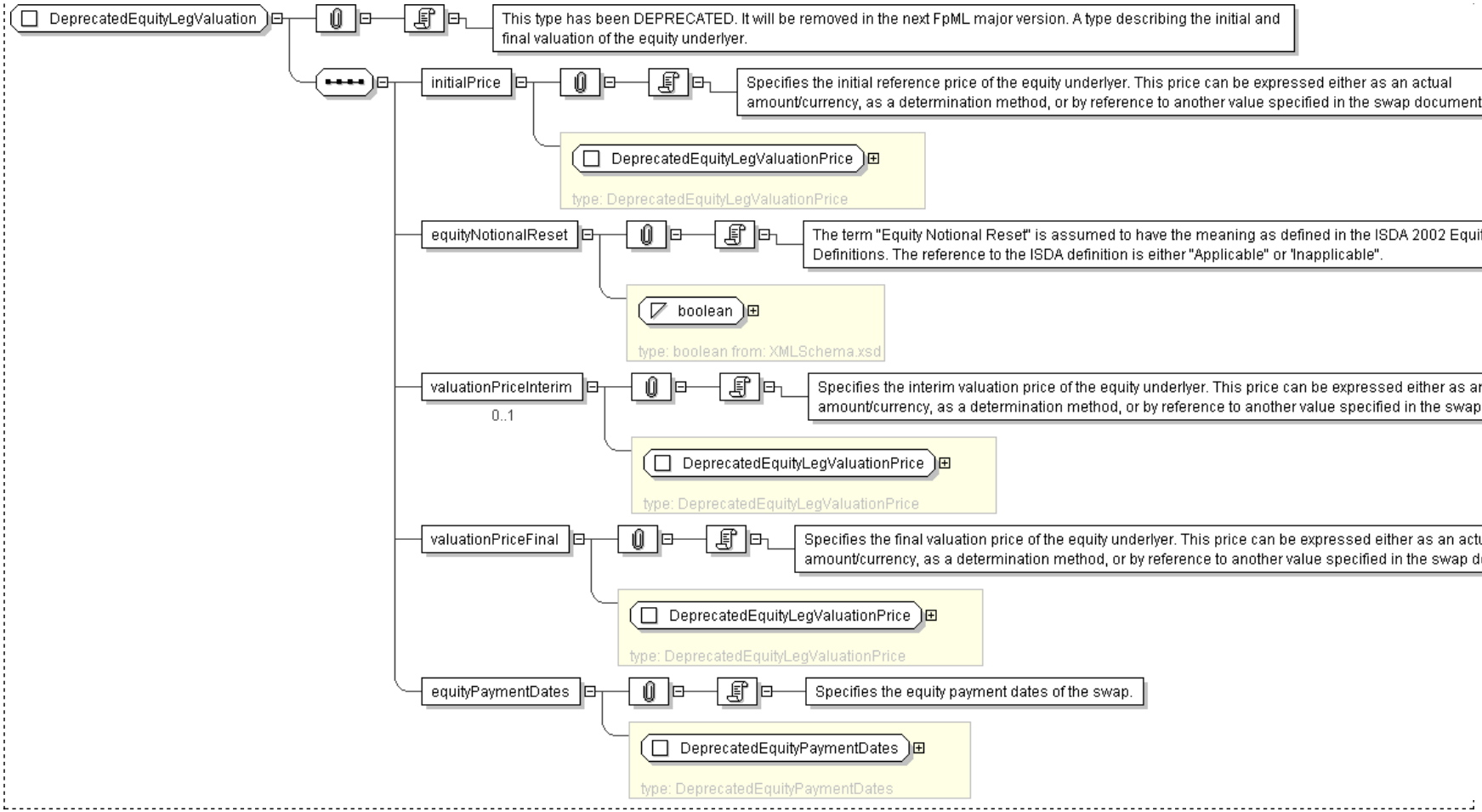
Super-types:	None
Sub-types:	None

Name	DeprecatedEquityLegValuation
Used by (from the same schema document)	Complex Type DeprecatedEquityLeg
Abstract	no
Documentation	This type has been DEPRECATED. It will be removed in the next FpML major version. A type describing the initial and final valuation of the equity underlyer.

XML Instance Representation

```
<...>
   <initialPrice> DeprecatedEquityLegValuationPrice </initialPrice> [1]
   'Specifies the initial reference price of the equity underlyer. This price can be expressed either as an actual amount/currency, as a
   determination method, or by reference to another value specified in the swap document.'

   <equityNotionalReset> xsd:boolean </equityNotionalReset> [1]
   'The term \"Equity Notional Reset\" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions. The reference to
   the ISDA definition is either \"Applicable\" or \'Inapplicable\".'

   <valuationPriceInterim> DeprecatedEquityLegValuationPrice </valuationPriceInterim> [0..1]
   'Specifies the interim valuation price of the equity underlyer. This price can be expressed either as an actual amount/currency, as a
   determination method, or by reference to another value specified in the swap document.'

   <valuationPriceFinal> DeprecatedEquityLegValuationPrice </valuationPriceFinal> [1]
   'Specifies the final valuation price of the equity underlyer. This price can be expressed either as an actual amount/currency, as a
   determination method, or by reference to another value specified in the swap document.'

   <equityPaymentDates> DeprecatedEquityPaymentDates </equityPaymentDates> [1]
   'Specifies the equity payment dates of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedEquityLegValuation" deprecated="true" deprecatedReason="It has been made more generic as ReturnLegValuation to cover
return swap type products.">
  <xsd:sequence>
    <xsd:element name="initialPrice" type=" DeprecatedEquityLegValuationPrice "/>
    <xsd:element name="equityNotionalReset" type=" xsd:boolean "/>
    <xsd:element name="valuationPriceInterim" type=" DeprecatedEquityLegValuationPrice " minOccurs="0"/>
    <xsd:element name="valuationPriceFinal" type=" DeprecatedEquityLegValuationPrice "/>
    <xsd:element name="equityPaymentDates" type=" DeprecatedEquityPaymentDates "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedEquityLegValuationPrice

[Table of contents]

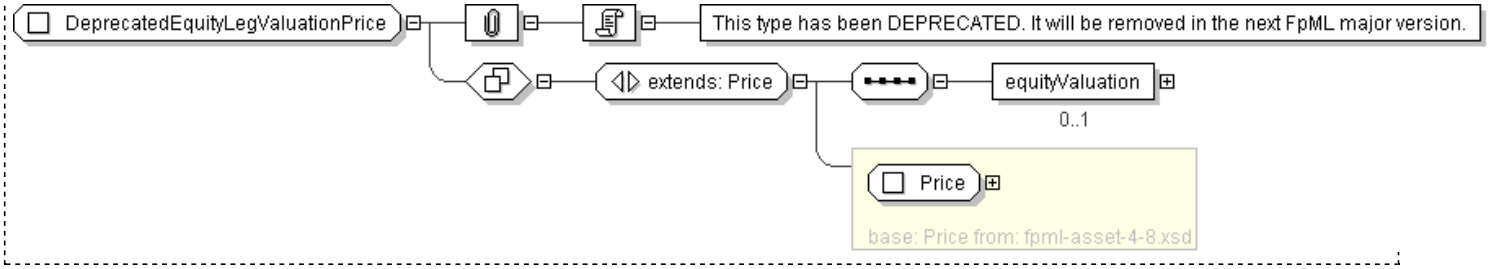
Super-types:	Price < DeprecatedEquityLegValuationPrice (by extension)
Sub-types:	None

Name	DeprecatedEquityLegValuationPrice
Used by (from the same schema document)	Complex Type DeprecatedEquityLegValuation , Complex Type DeprecatedEquityLegValuation , Complex Type DeprecatedEquityLegValuation
Abstract	no
Documentation	This type has been DEPRECATED. It will be removed in the next FpML major version.

XML Instance Representation

```
<...>
   <commission> Commission </commission> [0..1]
   'This optional component specifies the commission to be charged for executing the hedge
   transactions.'

   Start Choice [1]
      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      Start Group: EquityPrice.model [0..1]
         <grossPrice> ActualPrice </grossPrice> [0..1]
         'Specifies the price of the underlyer, before commissions.'

         <netPrice> ActualPrice </netPrice> [1]
         'Specifies the price of the underlyer, net of commissions.'

         <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
         'Specifies the accrued interest that are part of the dirty price in the case of a fixed income
         security or a convertible bond. Expressed in percentage of the notional.'

         <fxConversion> FxConversion </fxConversion> [0..1]
         'Specifies the currency conversion rate that applies to an amount. This rate can either be
         defined elsewhere in the document (case of a quanto swap), or explicitly described through this
         component.'

      End Group: EquityPrice.model
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'The href attribute value will be a pointer style reference to the element or component elsewhere
      in the document where the anchor amount is defined.'

      <grossPrice> ActualPrice </grossPrice> [0..1]
      'Specifies the price of the underlyer, before commissions.'

      <netPrice> ActualPrice </netPrice> [1]
      'Specifies the price of the underlyer, net of commissions.'

      <accruedInterestPrice> xsd:decimal </accruedInterestPrice> [0..1]
      'Specifies the accrued interest that are part of the dirty price in the case of a fixed income
      security or a convertible bond. Expressed in percentage of the notional.'

      <fxConversion> FxConversion </fxConversion> [0..1]
      'Specifies the currency conversion rate that applies to an amount. This rate can either be
      defined elsewhere in the document (case of a quanto swap), or explicitly described through this
      component.'

   End Choice
   <cleanNetPrice> xsd:decimal </cleanNetPrice> [0..1]
   'The net price excluding accrued interest. The \"Dirty Price\" for bonds is put in the \"netPrice\"
   element, which includes accrued interest. Thus netPrice - cleanNetPrice = accruedInterest. The
   currency and price expression for this field are the same as those for the (dirty) netPrice.'

   <quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..1]
   'Allows information about how the price was quoted to be provided.'

   <equityValuation> EquityValuation </equityValuation> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedEquityLegValuationPrice" deprecated="true" deprecatedReason="It has been
made more generic as ReturnLegValuationPrice to cover return swap type products.">
  <xsd:complexContent>
    <xsd:extension base=" Price ">
      <xsd:sequence>
        <xsd:element name="equityValuation" type=" EquityValuation " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeprecatedEquityPaymentDates

[Table of contents]

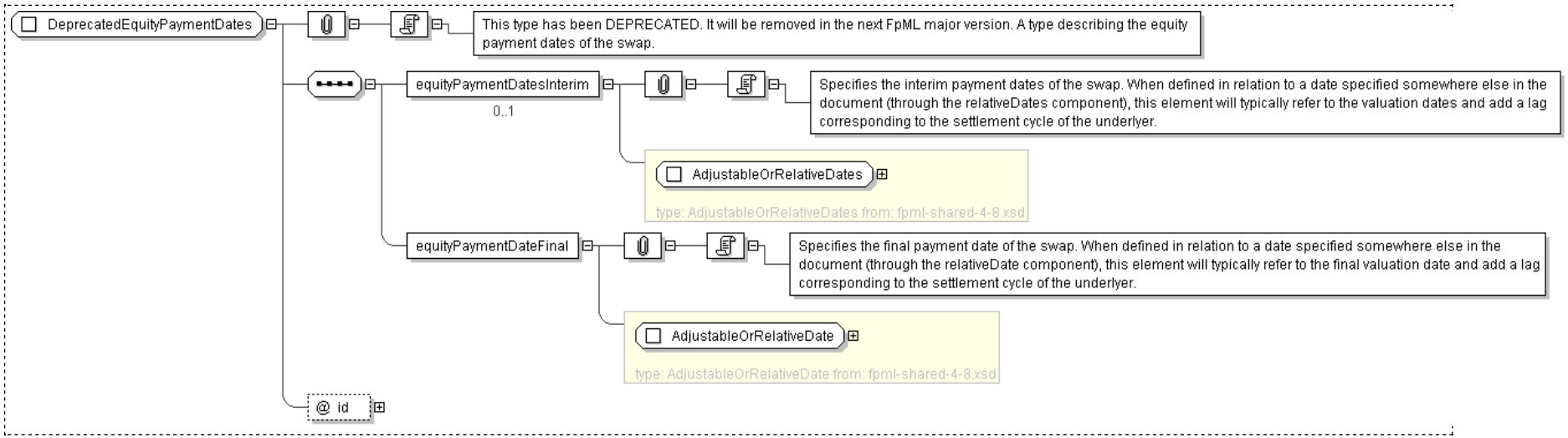
Super-types:	None
Sub-types:	None

Name	DeprecatedEquityPaymentDates
Used by (from the same schema document)	Complex Type DeprecatedEquityLegValuation
Abstract	no
Documentation	This type has been DEPRECATED. It will be removed in the next FpML major version. A type describing the equity payment dates of the swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <equityPaymentDatesInterim> AdjustableOrRelativeDates </equityPaymentDatesInterim> [0..1]
    'Specifies the interim payment dates of the swap. When defined in relation to a date specified somewhere else in the document (through the
    relativeDates component), this element will typically refer to the valuation dates and add a lag corresponding to the settlement cycle of the
    underlyer.'

   <equityPaymentDateFinal> AdjustableOrRelativeDate </equityPaymentDateFinal> [1]
    'Specifies the final payment date of the swap. When defined in relation to a date specified somewhere else in the document (through the relativeDate
    component), this element will typically refer to the final valuation date and add a lag corresponding to the settlement cycle of the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedEquityPaymentDates" deprecated="true" deprecatedReason="It has been made more generic as ReturnLegPaymentDates to cover
return swap type products.">
   <xsd:sequence>
      <xsd:element name="equityPaymentDatesInterim" type=" AdjustableOrRelativeDates " minOccurs="0"/>
      <xsd:element name="equityPaymentDateFinal" type=" AdjustableOrRelativeDate "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EquitySwapTransactionSupplement

[Table of contents]

Super-types:	ReturnSwapBase < EquitySwapTransactionSupplement (by extension)
Sub-types:	None

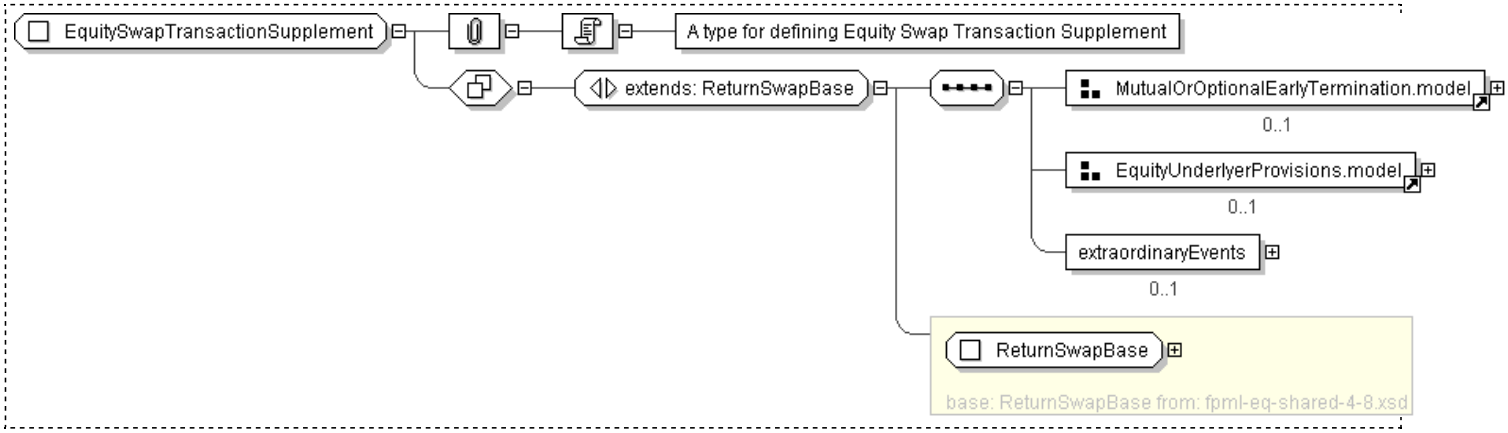
Name	EquitySwapTransactionSupplement
Used by (from the same schema document)	Element equitySwapTransactionSupplement
Abstract	no
Documentation	A type for defining Equity Swap Transaction Supplement

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding
   scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated
   with this element. Note that the domain values for this element are not strictly an enumerated list.'

   Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where
an implementor wishes to indicate who has manufactured the Swap through representing them as the Seller.
It may be removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and receives the
      rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed
      rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights
      defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article
      11.1 (a). In the case of FRAs this is the floating rate payer.'

   End Group: BuyerSeller.model
   <returnSwapLeg> ... </returnSwapLeg> [1..*]
   <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
   'This is used to document a Fully Funded Return Swap.'

   Start Group: MutualOrOptionalEarlyTermination.model [0..1]
      Start Choice [1]
         <mutualEarlyTermination> xsd:boolean </mutualEarlyTermination> [0..1]
         'Used for specifying whether the Mutual Early Termination Right that is detailed in the Master
         Confirmation will apply.'

         <optionalEarlyTermination> xsd:boolean </optionalEarlyTermination> [1]
         'A Boolean element used for specifying whether the Optional Early Termination clause detailed in
         the agreement will apply.'

         <breakFundingRecovery> xsd:boolean </breakFundingRecovery> [0..1]
         'A Boolean element used for specifying whether the Break Funding Recovery detailed in the agreement
         will apply.'

         Start Sequence [0..1]
            <breakFeeElection> FeeElectionEnum </breakFeeElection> [1]
            'Defines the fee type.'

            <breakFeeRate> NonNegativeDecimal </breakFeeRate> [0..1]
         End Sequence
      End Choice
   End Group: MutualOrOptionalEarlyTermination.model
   Start Group: EquityUnderlyerProvisions.model [0..1]
      Start Group: IndexAnnexFallback.model [0..1]
         Start Choice [1]
            <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
            'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index
            Annex is applicable to the transaction. This annex defines additional provisions which are
            applicable where an index is comprised of component securities that are traded on multiple
            exchanges.'

            <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
            'For an index option transaction, a flag to indicate whether a relevant Component Security Index
            Annex is applicable to the transaction.'
```

```
      End Choice
    End Group: IndexAnnexFallback.model
    <localJurisdiction> Country </localJurisdiction> [0..1]
    'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local
    taxes, which shall mean taxes, duties, and similar charges imposed by the taxing authority of the
    Local Jurisdiction If this element is not present Local Jurisdiction is Not Applicable.'

    <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
    'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine
    local taxes, which shall mean taxes, duties and similar charges that would be imposed by the taxing
    authority of the Country of Underlyer on a Hypothetical Broker Dealer assuming the Applicable Hedge
    Positions are held by its office in the Relevant Jurisdiction. If this element is not present
    Relevant Jurisdiction is Not Applicable.'

  End Group: EquityUnderlyerProvisions.model
  <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
  'Where the underlying is shares, specifies events affecting the issuer of those shares that may require
  the terms of the transaction to be adjusted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquitySwapTransactionSupplement">
  <xsd:complexContent>
    <xsd:extension base=" ReturnSwapBase ">
      <xsd:sequence>
        <xsd:group ref=" MutualOrOptionalEarlyTermination.model " minOccurs="0"/>
        <xsd:group ref=" EquityUnderlyerProvisions.model " minOccurs="0"/>
        <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: market
  - Element: pricingStructure
  - Element: pricingStructureValuation
- Global Definitions
  - Complex Type: AssetOrTermPointOrPricingStructureReference
  - Complex Type: BasicAssetValuation
  - Complex Type: DenominatorTerm
  - Complex Type: DerivativeCalculationMethod
  - Complex Type: DerivativeCalculationProcedure
  - Complex Type: DerivativeFormula
  - Complex Type: FormulaTerm
  - Complex Type: GenericDimension
  - Complex Type: InstrumentSet
  - Complex Type: Market
  - Complex Type: MarketReference
  - Complex Type: PerturbationType
  - Complex Type: PositionId
  - Complex Type: PricingDataPointCoordinate
  - Complex Type: PricingDataPointCoordinateReference
  - Complex Type: PricingInputReplacement
  - Complex Type: PricingInputType
  - Complex Type: PricingMethod
  - Complex Type: PricingParameterDerivative
  - Complex Type: PricingParameterDerivativeReference
  - Complex Type: PricingParameterShift
  - Complex Type: PricingStructureValuation
  - Complex Type: QuotedAssetSet
  - Complex Type: SensitivityDefinition
  - Complex Type: SensitivitySetDefinition
  - Complex Type: SensitivitySetDefinitionReference
  - Complex Type: TimeDimension
  - Complex Type: Valuation
  - Complex Type: ValuationReference
  - Complex Type: ValuationScenario
  - Complex Type: ValuationScenarioReference
  - Complex Type: WeightedPartialDerivative
  - Model Group: AnalyticDerivativeParameters.model
  - Model Group: ComputedDerivative.model
  - Model Group: DerivativeCalculationParameters.model
  - Model Group: FiniteDifferenceDerivativeParameters.model
  - Model Group: PositionIdAndVersion.model
  - Model Group: PricingCoordinateOrReference.model
  - Model Group: PricingInputDates.model
  - Model Group: PricingStructureIndex.model
  - Model Group: SensitivityDescription.model
  - Model Group: SubstitutionDerivativeParameters.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.

Schema Composition	• This schema includes components from the following schema document(s):
	  ○ fpml-doc-4-8.xsd
	  ○ fpml-asset-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-doc-4-8.xsd"/>
   <xsd:include schemaLocation="fpml-asset-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress
Schema Component Type — Schema Component Name

Super-types:	Address < AusAddress (by extension)
Sub-types:	• QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space

characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

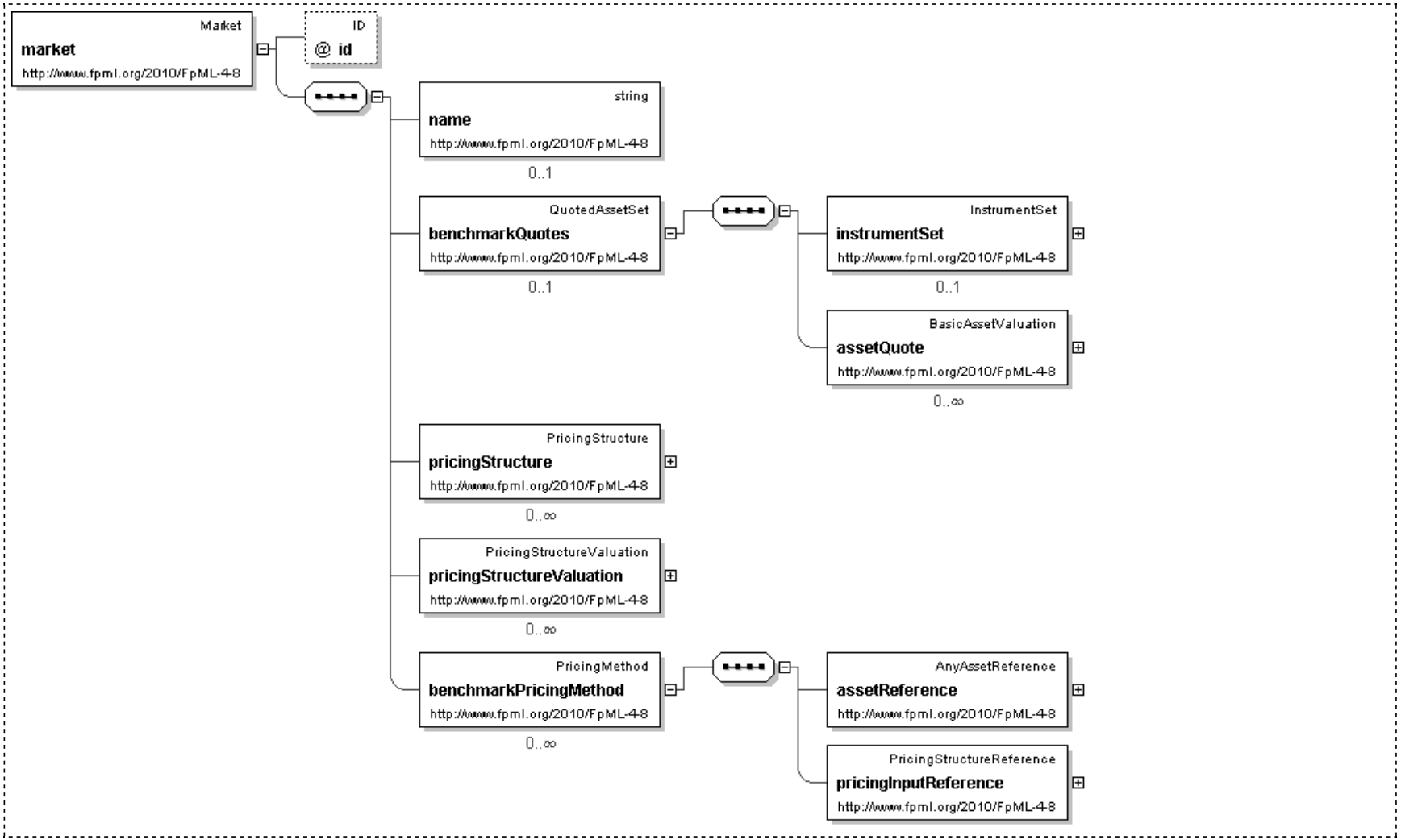
XML Schema Documentation

Element: market

[Table of contents]

Name	market
Type	Market
Nillable	no
Abstract	no
Documentation	This is a global element used for creating global types. It holds Market information, e.g. curves, surfaces, quotes, etc.

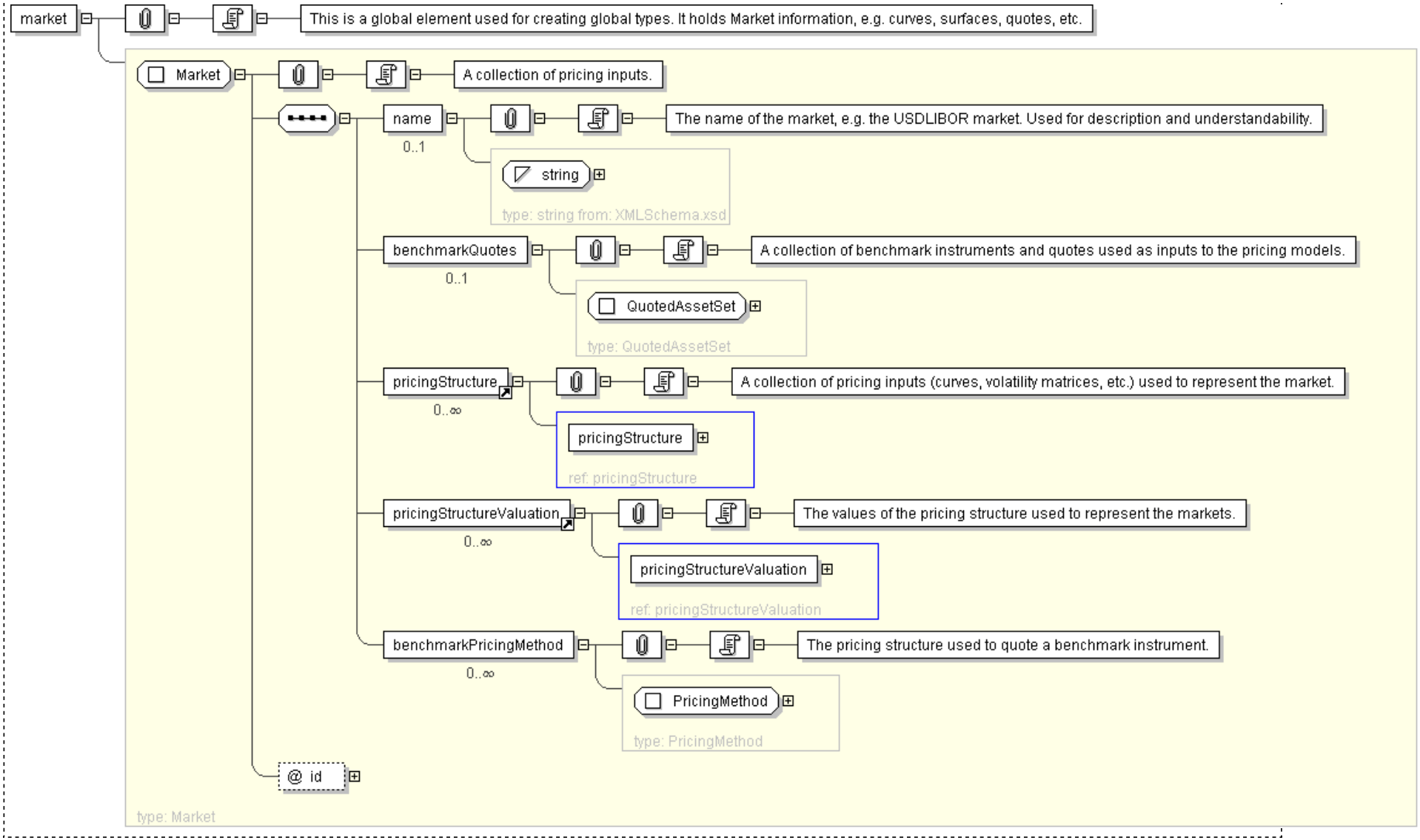
Logical Diagram



XML Instance Representation

```
<market
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The name of the market, e.g. the USDLIBOR market. Used for description and understandability.'

   <benchmarkQuotes> QuotedAssetSet </benchmarkQuotes> [0..1]
   'A collection of benchmark instruments and quotes used as inputs to the pricing models.'

   <pricingStructure> ... </pricingStructure> [0..*]
   'A collection of pricing inputs (curves, volatility matrices, etc.) used to represent the market.'

   <pricingStructureValuation> ... </pricingStructureValuation> [0..*]
   'The values of the pricing structure used to represent the markets.'

   <benchmarkPricingMethod> PricingMethod </benchmarkPricingMethod> [0..*]
   'The pricing structure used to quote a benchmark instrument.'

</market>
```

Diagram



Schema Component Representation

```
<xsd:element name="market" type=" Market "/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

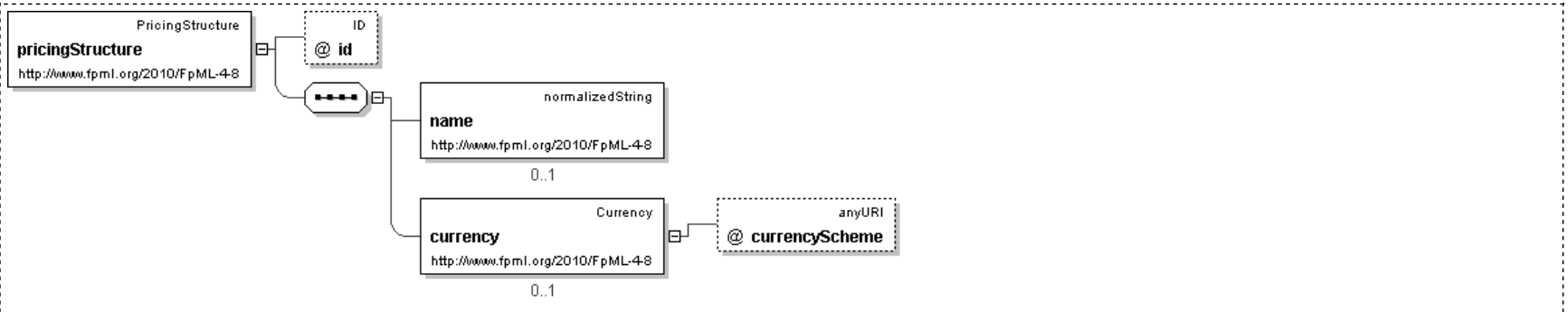
XML Schema Documentation

Element: pricingStructure

[Table of contents]

Name	pricingStructure
Used by (from the same schema document)	Complex Type Market
Type	PricingStructure
Nillable	no
Abstract	yes

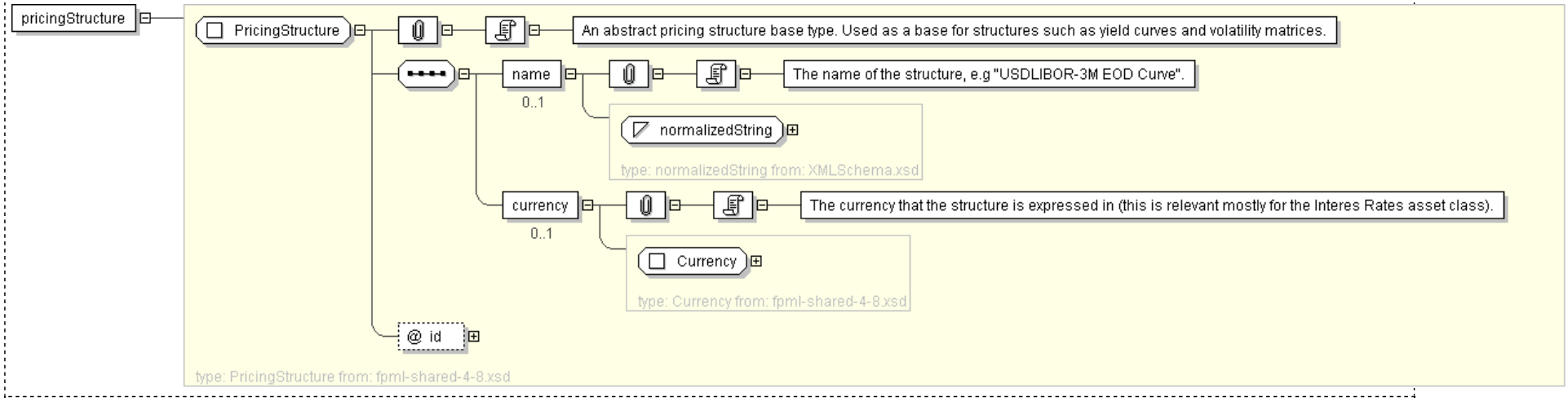
Logical Diagram



XML Instance Representation

```
<pricingStructure
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
   'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
   'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates asset class).'

</pricingStructure>
```

Diagram



Schema Component Representation

```
<xsd:element name="pricingStructure" type=" PricingStructure " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

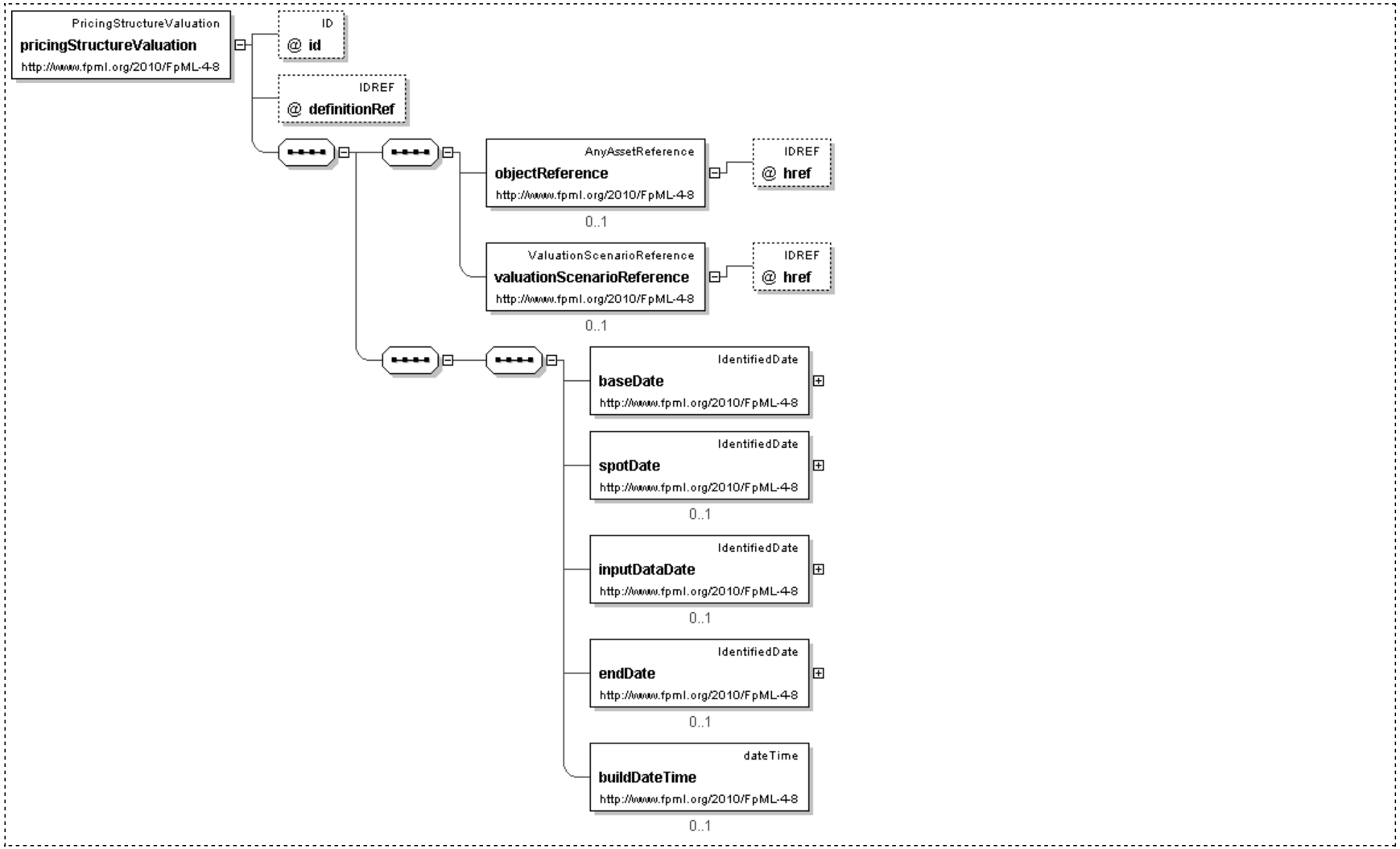
XML Schema Documentation

Element: pricingStructureValuation

[Table of contents]

Name	pricingStructureValuation
Used by (from the same schema document)	Complex Type Market
Type	PricingStructureValuation
Nillable	no
Abstract	yes

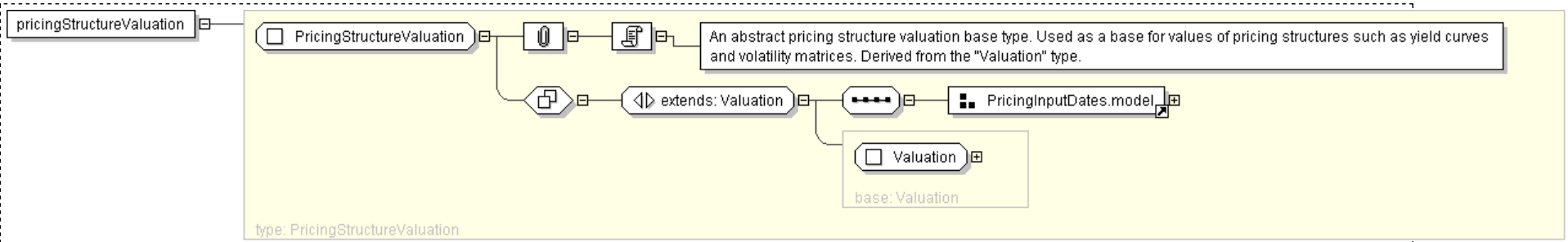
Logical Diagram



XML Instance Representation

```
<pricingStructureValuation
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a ValuationSet, this value is
   optional and is defaulted from the ValuationSet. If this value occurs in both places, the lower level value (i.e. the one here) overrides
   that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference between the baseDate and
   the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the baseDate, but sometimes the
   pricing input may be \"rolled forward\", in which input data from one date is used to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

</pricingStructureValuation>
```

Diagram



Schema Component Representation

```
<xsd:element name="pricingStructureValuation" type=" PricingStructureValuation " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AnalyticDerivativeParameters.model

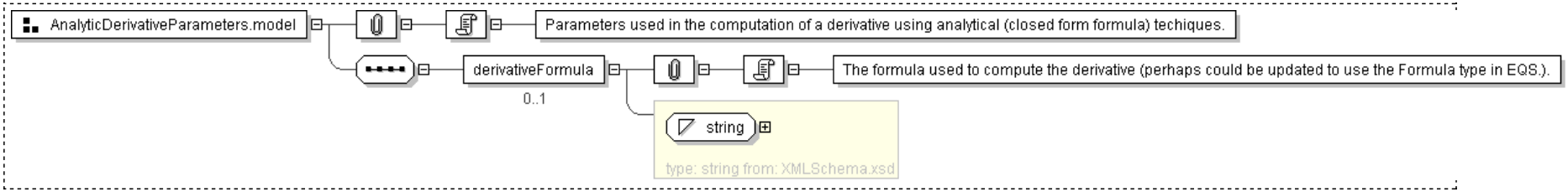
[Table of contents]

Name	AnalyticDerivativeParameters.model
Used by (from the same schema document)	Model Group DerivativeCalculationParameters.model
Documentation	Parameters used in the computation of a derivative using analytical (closed form formula) techiques.

XML Instance Representation

```
<derivativeFormula> xsd:string </derivativeFormula> [0..1]
'The formula used to compute the derivative (perhaps could be updated to use the Formula type in EQS.).'
```

Diagram



Schema Component Representation

```
<xsd:group name="AnalyticDerivativeParameters.model">
  <xsd:sequence>
    <xsd:element name="derivativeFormula" type=" xsd:string " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: ComputedDerivative.model

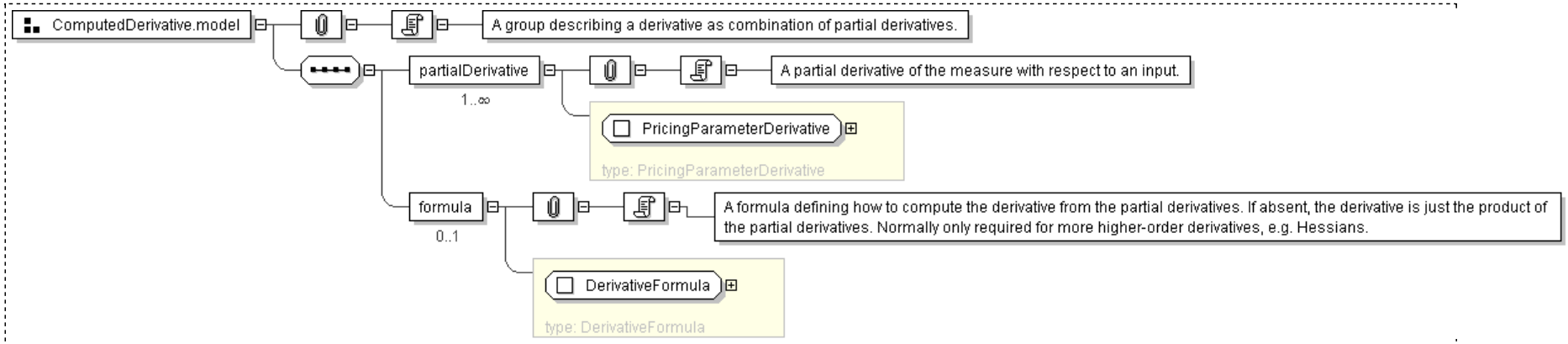
[Table of contents]

Name	ComputedDerivative.model
Used by (from the same schema document)	Complex Type SensitivityDefinition
Documentation	A group describing a derivative as combination of partial derivatives.

XML Instance Representation

```
<partialDerivative> PricingParameterDerivative </partialDerivative> [1..*]
'A partial derivative of the measure with respect to an input.'

<formula> DerivativeFormula </formula> [0..1]
'A formula defining how to compute the derivative from the partial derivatives. If absent, the derivative is just the product of the
partial derivatives. Normally only required for more higher-order derivatives, e.g. Hessians.'
```

Diagram



Schema Component Representation

```
<xsd:group name="ComputedDerivative.model">
  <xsd:sequence>
    <xsd:element name="partialDerivative" type=" PricingParameterDerivative " maxOccurs="unbounded"/>
    <xsd:element name="formula" type=" DerivativeFormula " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: DerivativeCalculationParameters.model

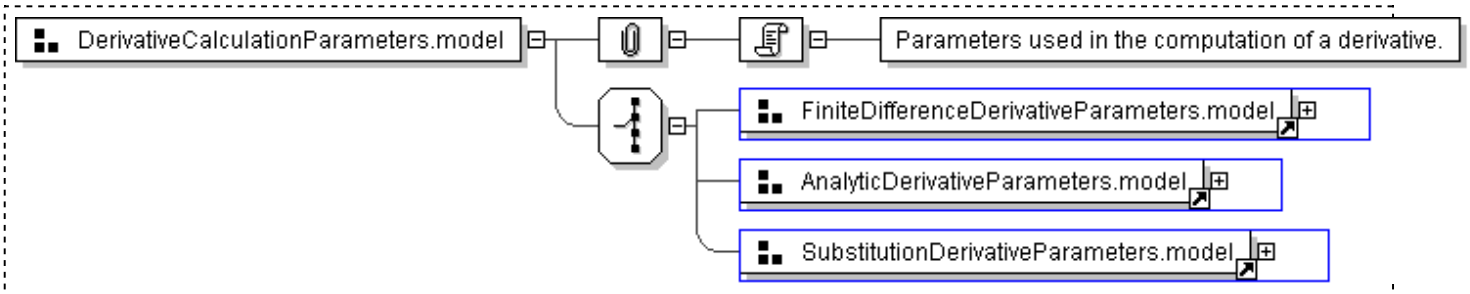
[Table of contents]

Name	DerivativeCalculationParameters.model
Used by (from the same schema document)	Complex Type DerivativeCalculationProcedure
Documentation	Parameters used in the computation of a derivative.

XML Instance Representation

```
Start Choice [1]
   <perturbationAmount> xsd:decimal </perturbationAmount> [0..1]
   'The size and direction of the perturbation used to compute the derivative, e.g.
   0.0001 = 1 bp.'

   <averaged> xsd:boolean </averaged> [1]
   'The value is calculated by perturbing by the perturbationAmount and then the
   negative of the perturbationAmount and then averaging the two values (i.e. the
   value is half of the difference between perturbing up and perturbing down).'

   <perturbationType> PerturbationType </perturbationType> [0..1]
   'The type of perturbation, if any, used to compute the derivative (Absolute vs
   Relative).'

   <derivativeFormula> xsd:string </derivativeFormula> [0..1]
   'The formula used to compute the derivative (perhaps could be updated to use the
   Formula type in EQS.).'

   <replacementMarketInput> PricingStructureReference </replacementMarketInput> [1]
   'A reference to the replacement version of the market input, e.g. a bumped yield
   curve.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="DerivativeCalculationParameters.model">
   <xsd:choice>
      <xsd:group ref=" FiniteDifferenceDerivativeParameters.model "/>
      <xsd:group ref=" AnalyticDerivativeParameters.model "/>
      <xsd:group ref=" SubstitutionDerivativeParameters.model "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: FiniteDifferenceDerivativeParameters.model

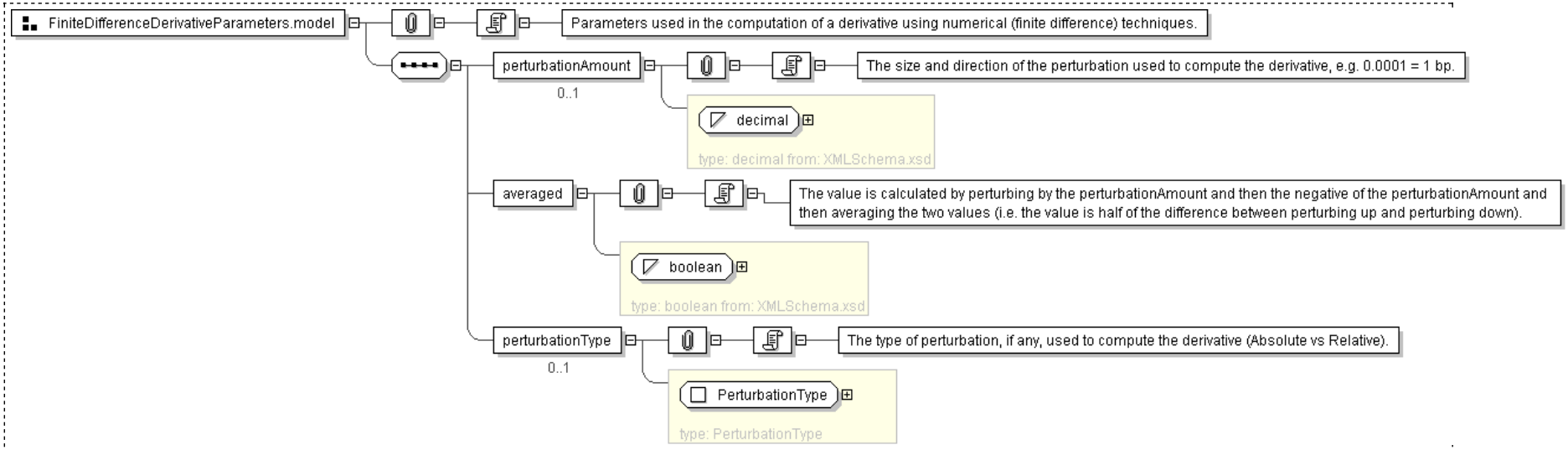
[Table of contents]

Name	FiniteDifferenceDerivativeParameters.model
Used by (from the same schema document)	Model Group DerivativeCalculationParameters.model
Documentation	Parameters used in the computation of a derivative using numerical (finite difference) techniques.

XML Instance Representation

```
<perturbationAmount> xsd:decimal </perturbationAmount> [0..1]
'The size and direction of the perturbation used to compute the derivative, e.g. 0.0001 = 1 bp.'

<averaged> xsd:boolean </averaged> [1]
'The value is calculated by perturbing by the perturbationAmount and then the negative of the perturbationAmount and then averaging the two values
(i.e. the value is half of the difference between perturbing up and perturbing down).'

<perturbationType> PerturbationType </perturbationType> [0..1]
'The type of perturbation, if any, used to compute the derivative (Absolute vs Relative).'
```

Diagram



Schema Component Representation

```
<xsd:group name="FiniteDifferenceDerivativeParameters.model">
  <xsd:sequence>
    <xsd:element name="perturbationAmount" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="averaged" type=" xsd:boolean "/>
    <xsd:element name="perturbationType" type=" PerturbationType " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PositionIdAndVersion.model

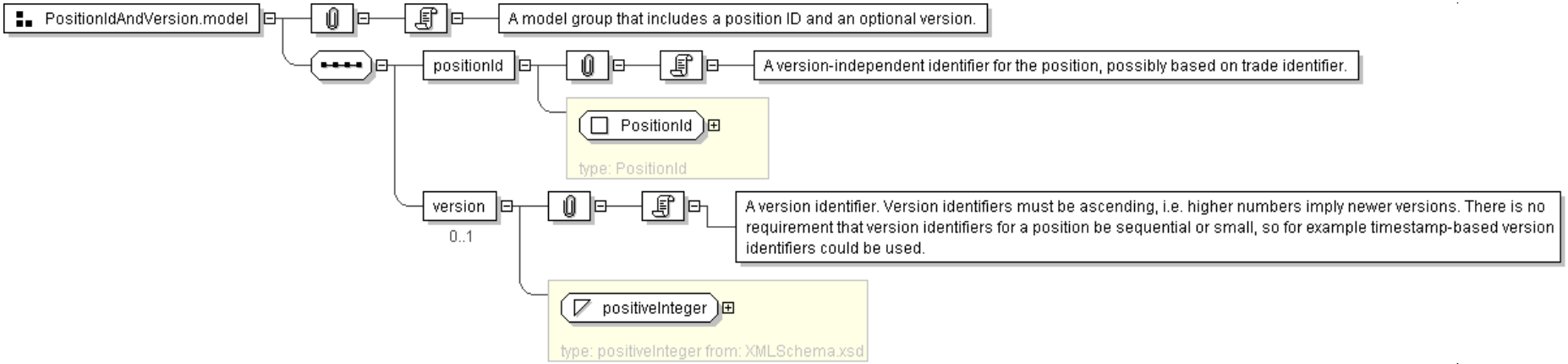
[Table of contents]

Name	PositionIdAndVersion.model
Documentation	A model group that includes a position ID and an optional version.

XML Instance Representation

```
<positionId> PositionId </positionId> [1]
'A version-independent identifier for the position, possibly based on trade identifier.'

<version> xsd:positiveInteger </version> [0..1]
'A version identifier. Version identifiers must be ascending, i.e. higher numbers imply newer versions. There is no requirement that
version identifiers for a position be sequential or small, so for example timestamp-based version identifiers could be used.'
```

Diagram



Schema Component Representation

```
<xsd:group name="PositionIdAndVersion.model">
  <xsd:sequence>
    <xsd:element name="positionId" type=" PositionId "/>
    <xsd:element name="version" type=" xsd:positiveInteger " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PricingCoordinateOrReference.model

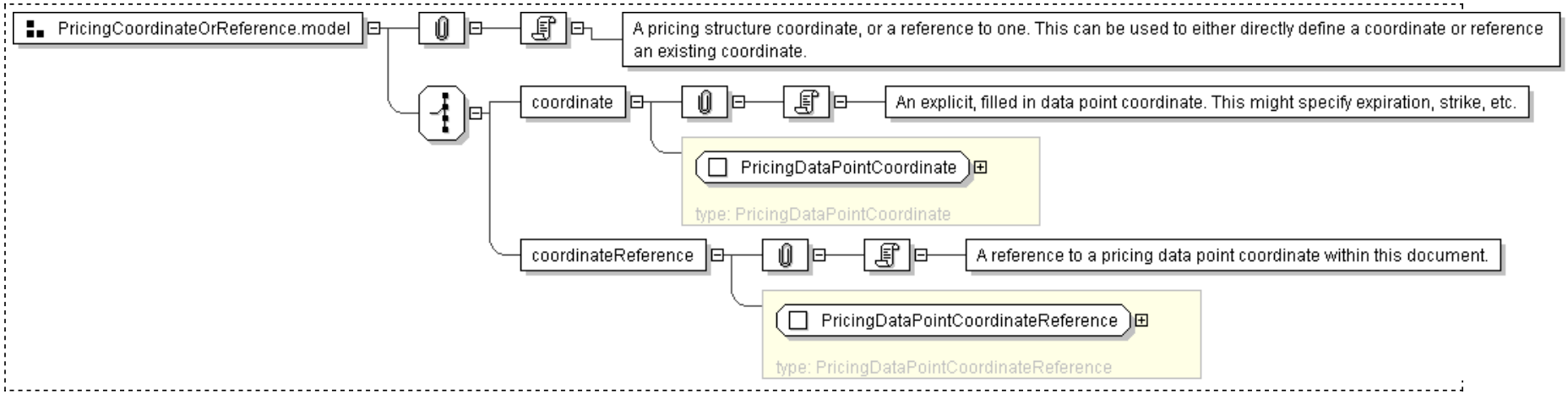
[Table of contents]

Name	PricingCoordinateOrReference.model
Used by (from the same schema document)	Model Group SensitivityDescription.model
Documentation	A pricing structure coordinate, or a reference to one. This can be used to either directly define a coordinate or reference an existing coordinate.

XML Instance Representation

```
Start Choice [1]
   <coordinate> PricingDataPointCoordinate </coordinate> [1]
    'An explicit, filled in data point coordinate. This might specify expiration, strike, etc.'

   <coordinateReference> PricingDataPointCoordinateReference </coordinateReference> [1]
    'A reference to a pricing data point coordinate within this document.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="PricingCoordinateOrReference.model">
   <xsd:choice>
      <xsd:element name="coordinate" type=" PricingDataPointCoordinate "/>
      <xsd:element name="coordinateReference" type=" PricingDataPointCoordinateReference "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PricingInputDates.model

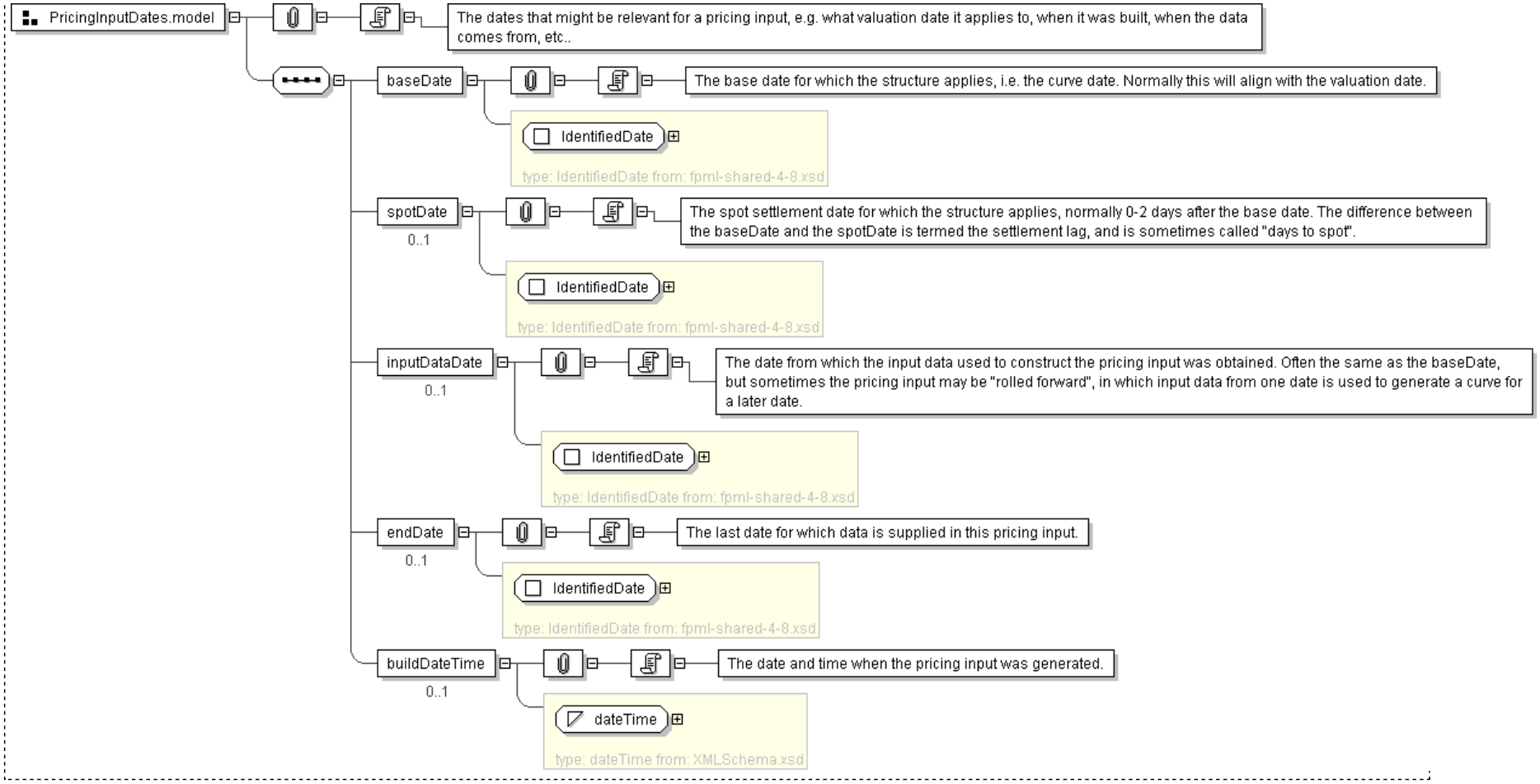
[Table of contents]

Name	PricingInputDates.model
Used by (from the same schema document)	Complex Type PricingStructureValuation
Documentation	The dates that might be relevant for a pricing input, e.g. what valuation date it applies to, when it was built, when the data comes from, etc..

XML Instance Representation

```
<baseDate> IdentifiedDate </baseDate> [1]
'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation date.'

<spotDate> IdentifiedDate </spotDate> [0..1]
'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference between the baseDate and the
spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

<inputDataDate> IdentifiedDate </inputDataDate> [0..1]
'The date from which the input data used to construct the pricing input was obtained. Often the same as the baseDate, but sometimes the
pricing input may be \"rolled forward\", in which input data from one date is used to generate a curve for a later date.'

<endDate> IdentifiedDate </endDate> [0..1]
'The last date for which data is supplied in this pricing input.'

<buildDateTime> xsd:dateTime </buildDateTime> [0..1]
'The date and time when the pricing input was generated.'
```

Diagram



Schema Component Representation

```
<xsd:group name="PricingInputDates.model">
  <xsd:sequence>
    <xsd:element name="baseDate" type=" IdentifiedDate "/>
    <xsd:element name="spotDate" type=" IdentifiedDate " minOccurs="0"/>
    <xsd:element name="inputDataDate" type=" IdentifiedDate " minOccurs="0"/>
    <xsd:element name="endDate" type=" IdentifiedDate " minOccurs="0"/>
    <xsd:element name="buildDateTime" type=" xsd:dateTime " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PricingStructureIndex.model

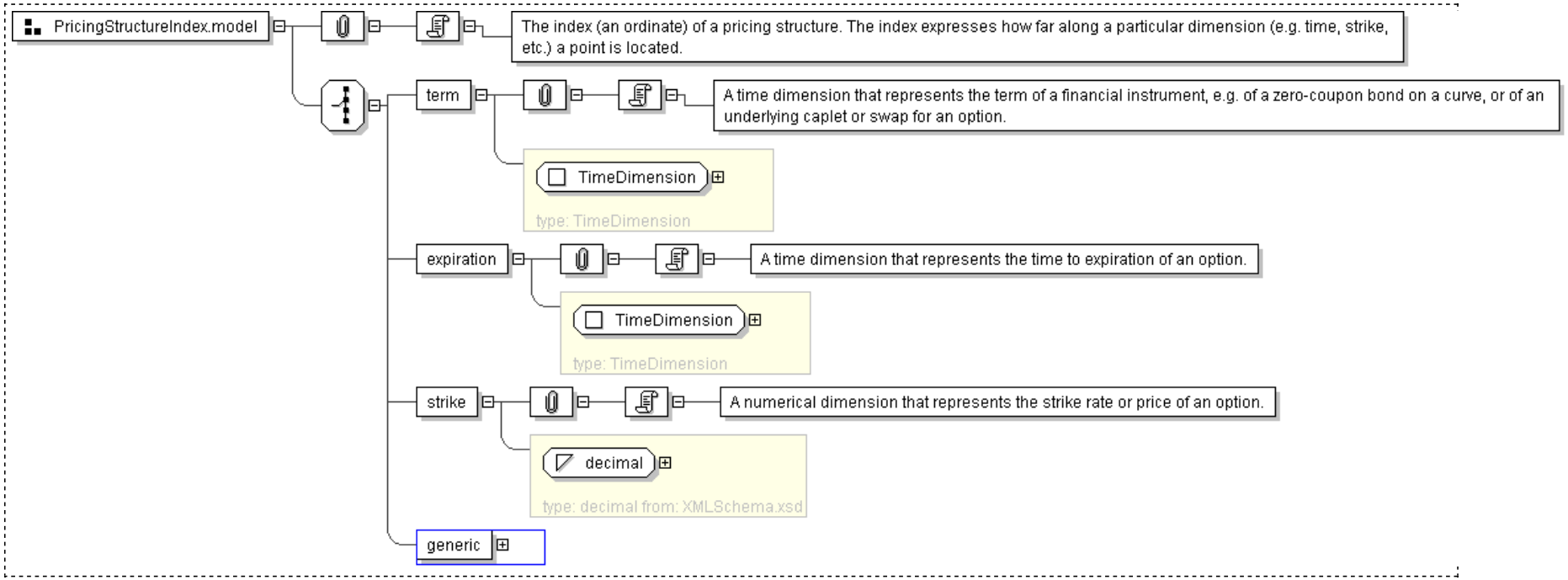
[Table of contents]

Name	PricingStructureIndex.model
Used by (from the same schema document)	Complex Type PricingDataPointCoordinate
Documentation	The index (an ordinate) of a pricing structure. The index expresses how far along a particular dimension (e.g. time, strike, etc.) a point is located.

XML Instance Representation

```
Start Choice [1]
   <term> TimeDimension </term> [1]
   'A time dimension that represents the term of a financial instrument, e.g. of a zero-coupon bond on a curve, or of an underlying
   caplet or swap for an option.'

   <expiration> TimeDimension </expiration> [1]
   'A time dimension that represents the time to expiration of an option.'

   <strike> xsd:decimal </strike> [1]
   'A numerical dimension that represents the strike rate or price of an option.'

   <generic> GenericDimension </generic> [1]
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="PricingStructureIndex.model">
```

```
   <xsd:choice>
      <xsd:element name="term" type=" TimeDimension "/>
      <xsd:element name="expiration" type=" TimeDimension "/>
      <xsd:element name="strike" type=" xsd:decimal "/>
      <xsd:element name="generic" type=" GenericDimension "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: SensitivityDescription.model

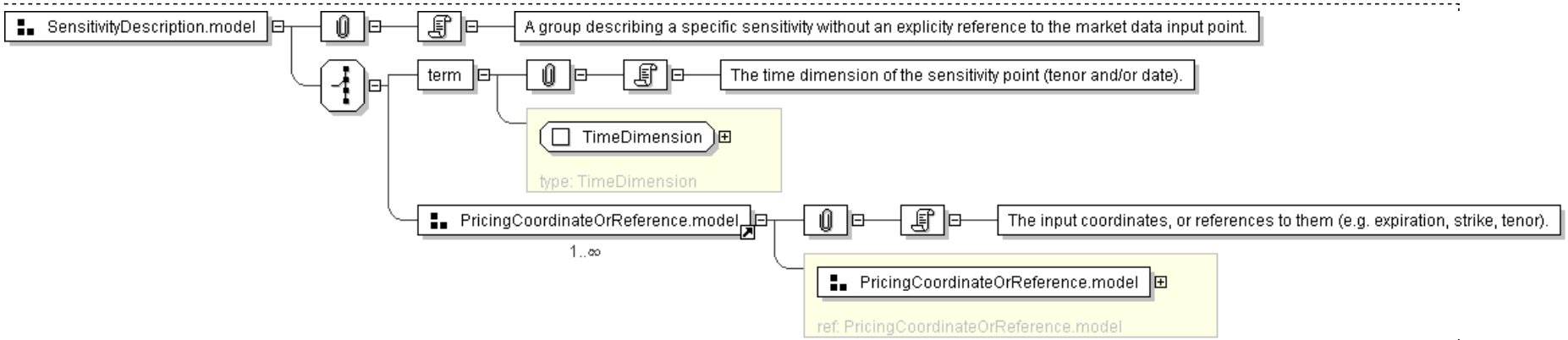
[Table of contents]

Name	SensitivityDescription.model
Used by (from the same schema document)	Complex Type SensitivityDefinition
Documentation	A group describing a specific sensitivity without an explicity reference to the market data input point.

XML Instance Representation

```
Start Choice [1]
   <term> TimeDimension </term> [1]
   'The time dimension of the sensitivity point (tenor and/or date).'

   Start Group: PricingCoordinateOrReference.model [1..*]
'The input coordinates, or references to them (e.g. expiration, strike, tenor).'


      Start Choice [1]
         <coordinate> PricingDataPointCoordinate </coordinate> [1]
         'An explicit, filled in data point coordinate. This might specify expiration, strike, etc.'

         <coordinateReference> PricingDataPointCoordinateReference </coordinateReference> [1]
         'A reference to a pricing data point coordinate within this document.'

      End Choice
   End Group: PricingCoordinateOrReference.model
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="SensitivityDescription.model">
   <xsd:choice>
      <xsd:element name="term" type=" TimeDimension "/>
      <xsd:group ref=" PricingCoordinateOrReference.model " maxOccurs="unbounded"/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: SubstitutionDerivativeParameters.model

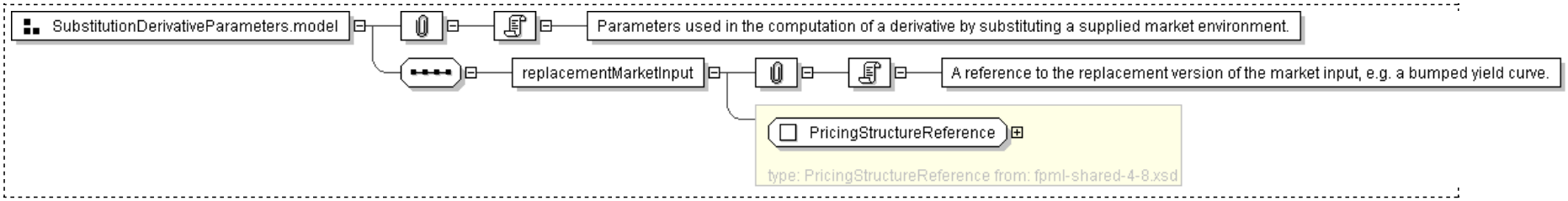
[Table of contents]

Name	SubstitutionDerivativeParameters.model
Used by (from the same schema document)	Model Group DerivativeCalculationParameters.model
Documentation	Parameters used in the computation of a derivative by substituting a supplied market environment.

XML Instance Representation

```
<replacementMarketInput> PricingStructureReference </replacementMarketInput> [1]
'A reference to the replacement version of the market input, e.g. a bumped yield curve.'
```

Diagram



Schema Component Representation

```
<xsd:group name="SubstitutionDerivativeParameters.model">
  <xsd:sequence>
    <xsd:element name="replacementMarketInput" type=" PricingStructureReference "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssetOrTermPointOrPricingStructureReference

[Table of contents]

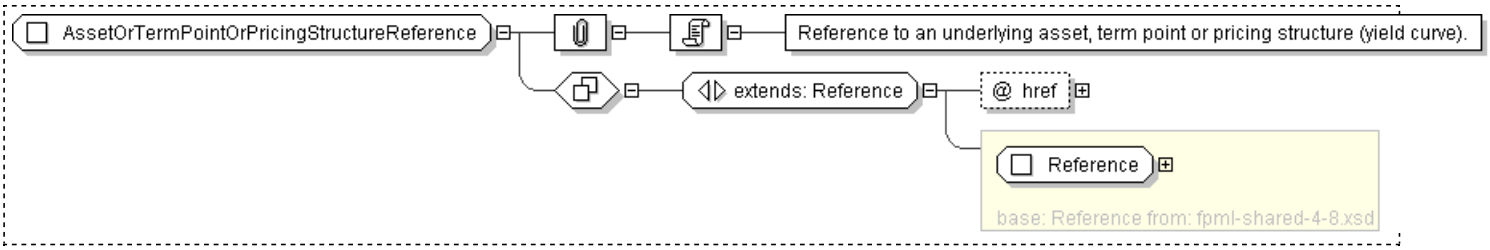
Super-types:	Reference < AssetOrTermPointOrPricingStructureReference (by extension)
Sub-types:	None

Name	AssetOrTermPointOrPricingStructureReference
Used by (from the same schema document)	Complex Type PricingParameterDerivative , Complex Type PricingParameterShift
Abstract	no
Documentation	Reference to an underlying asset, term point or pricing structure (yield curve).

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssetOrTermPointOrPricingStructureReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BasicAssetValuation

[Table of contents]

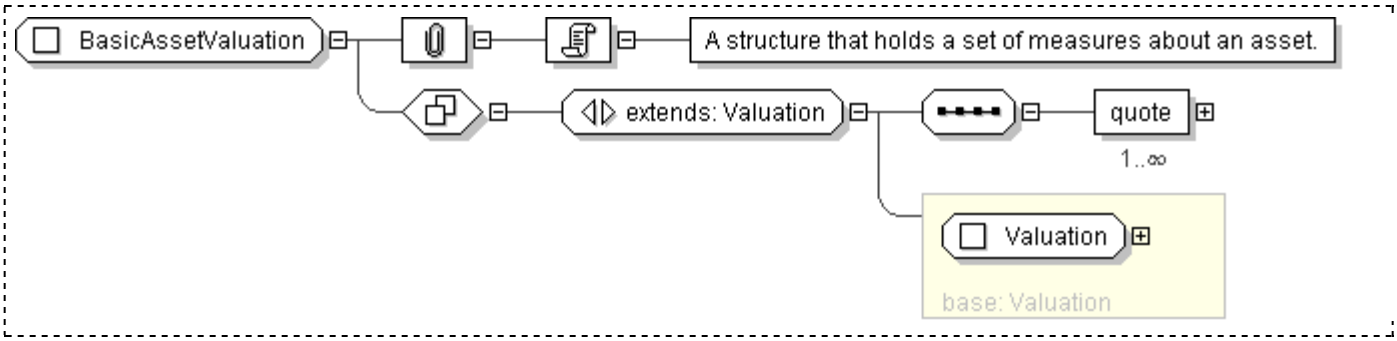
Super-types:	Valuation < BasicAssetValuation (by extension)
Sub-types:	None

Name	BasicAssetValuation
Used by (from the same schema document)	Complex Type QuotedAssetSet
Abstract	no
Documentation	A structure that holds a set of measures about an asset.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference
   </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the
   Valuation occurs within a ValuationSet, this value is optional and is defaulted
   from the ValuationSet. If this value occurs in both places, the lower level
   value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <quote> BasicQuotation </quote> [1..*]
   'One or more numerical measures relating to the asset, possibly together with
   sensitivities of that measure to pricing inputs'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BasicAssetValuation">
  <xsd:complexContent>
    <xsd:extension base=" Valuation ">
      <xsd:sequence>
        <xsd:element name="quote" type=" BasicQuotation " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
```

```
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DenominatorTerm

[Table of contents]

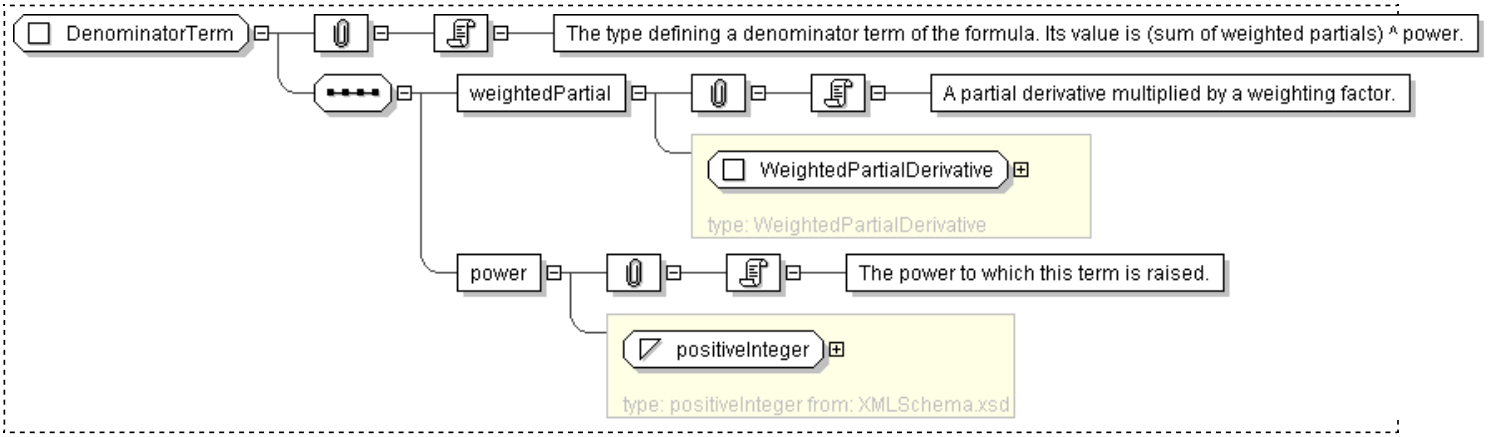
Super-types:	None
Sub-types:	None

Name	DenominatorTerm
Used by (from the same schema document)	Complex Type DerivativeFormula
Abstract	no
Documentation	The type defining a denominator term of the formula. Its value is (sum of weighted partials) ^ power.

XML Instance Representation

```
<...>
   <weightedPartial> WeightedPartialDerivative </weightedPartial> [1]
    'A partial derivative multiplied by a weighting factor.'

   <power> xsd:positiveInteger </power> [1]
    'The power to which this term is raised.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DenominatorTerm">
   <xsd:sequence>
      <xsd:element name="weightedPartial" type=" WeightedPartialDerivative "/>
      <xsd:element name="power" type=" xsd:positiveInteger "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DerivativeCalculationMethod

[Table of contents]

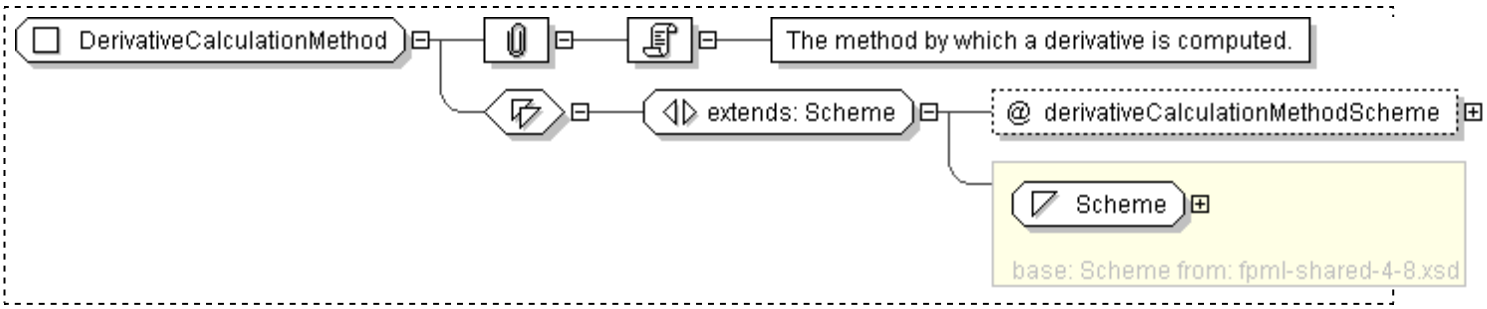
Super-types:	Scheme < DerivativeCalculationMethod (by extension)
Sub-types:	None

Name	DerivativeCalculationMethod
Used by (from the same schema document)	Complex Type DerivativeCalculationProcedure
Abstract	no
Documentation	The method by which a derivative is computed.

XML Instance Representation

```
<...
 derivativeCalculationMethodScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DerivativeCalculationMethod">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="derivativeCalculationMethodScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/derivative-calculation-method"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DerivativeCalculationProcedure

[Table of contents]

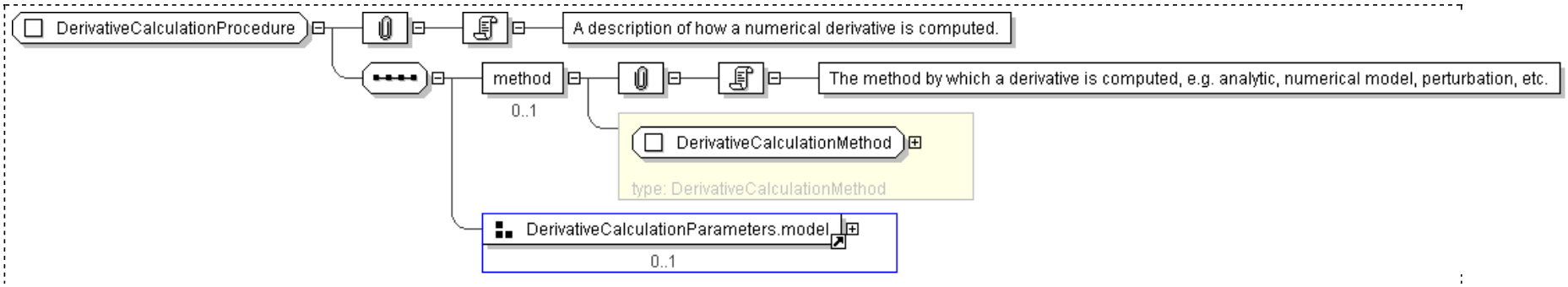
Super-types:	None
Sub-types:	None

Name	DerivativeCalculationProcedure
Used by (from the same schema document)	Complex Type PricingParameterDerivative , Complex Type SensitivitySetDefinition
Abstract	no
Documentation	A description of how a numerical derivative is computed.

XML Instance Representation

```
<...>
   <method> DerivativeCalculationMethod </method> [0..1]
    'The method by which a derivative is computed, e.g. analytic, numerical model, perturbation, etc.'

   Start Group: DerivativeCalculationParameters.model [0..1]
      Start Choice [1]
         <perturbationAmount> xsd:decimal </perturbationAmount> [0..1]
          'The size and direction of the perturbation used to compute the derivative, e.g. 0.0001 = 1 bp.'

         <averaged> xsd:boolean </averaged> [1]
          'The value is calculated by perturbing by the perturbationAmount and then the negative of the perturbationAmount and
          then averaging the two values (i.e. the value is half of the difference between perturbing up and perturbing down).'

         <perturbationType> PerturbationType </perturbationType> [0..1]
          'The type of perturbation, if any, used to compute the derivative (Absolute vs Relative).'

         <derivativeFormula> xsd:string </derivativeFormula> [0..1]
          'The formula used to compute the derivative (perhaps could be updated to use the Formula type in EQS.).'

         <replacementMarketInput> PricingStructureReference </replacementMarketInput> [1]
          'A reference to the replacement version of the market input, e.g. a bumped yield curve.'

      End Choice
   End Group: DerivativeCalculationParameters.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DerivativeCalculationProcedure">
  <xsd:sequence>
    <xsd:element name="method" type=" DerivativeCalculationMethod " minOccurs="0"/>
    <xsd:group ref=" DerivativeCalculationParameters.model " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DerivativeFormula

[Table of contents]

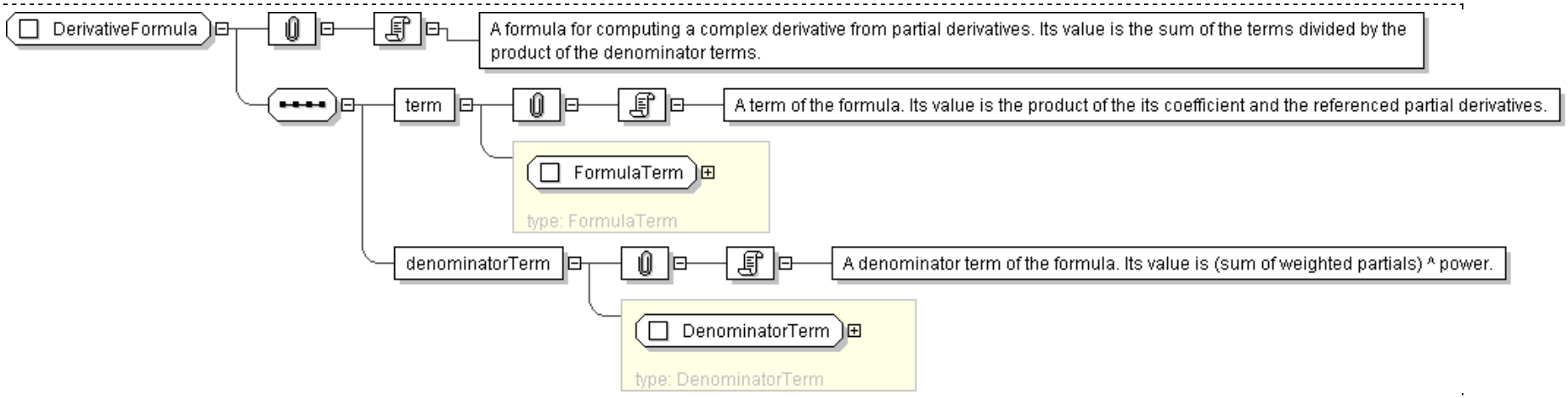
Super-types:	None
Sub-types:	None

Name	DerivativeFormula
Used by (from the same schema document)	Model Group ComputedDerivative.model
Abstract	no
Documentation	A formula for computing a complex derivative from partial derivatives. Its value is the sum of the terms divided by the product of the denominator terms.

XML Instance Representation

```
<...>
   <term> FormulaTerm </term> [1]
   'A term of the formula. Its value is the product of the its coefficient and the referenced partial derivatives.'

   <denominatorTerm> DenominatorTerm </denominatorTerm> [1]
   'A denominator term of the formula. Its value is (sum of weighted partials) ^ power.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DerivativeFormula">
   <xsd:sequence>
      <xsd:element name="term" type=" FormulaTerm "/>
      <xsd:element name="denominatorTerm" type=" DenominatorTerm "/>
```

```
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FormulaTerm

[Table of contents]

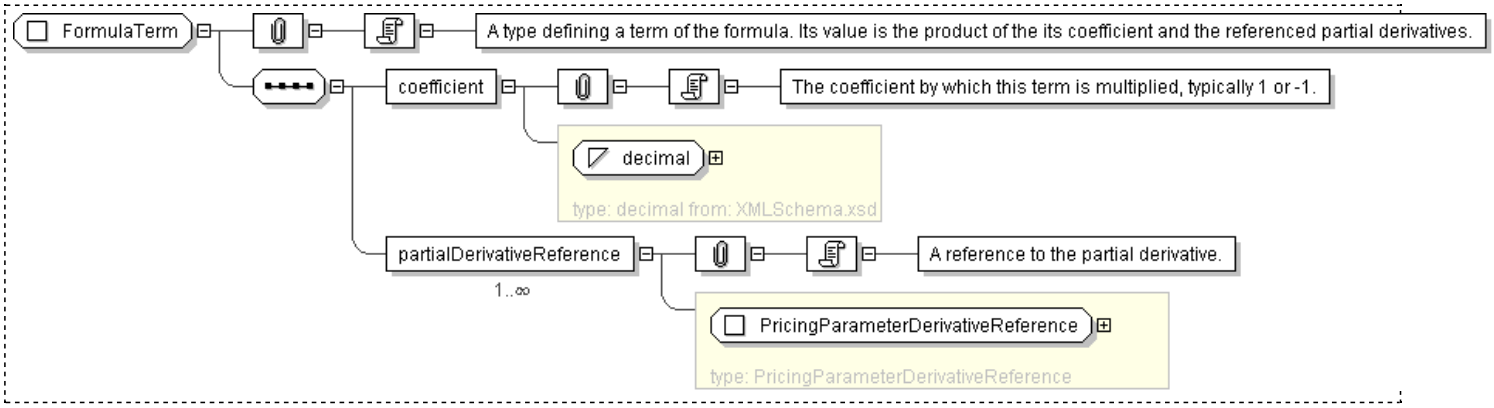
Super-types:	None
Sub-types:	None

Name	FormulaTerm
Used by (from the same schema document)	Complex Type DerivativeFormula
Abstract	no
Documentation	A type defining a term of the formula. Its value is the product of the its coefficient and the referenced partial derivatives.

XML Instance Representation

```
<...>
  <coefficient> xsd:decimal </coefficient> [1]
  'The coefficient by which this term is multiplied, typically 1 or -1.'

  <partialDerivativeReference> PricingParameterDerivativeReference </partialDerivativeReference> [1..*]
  'A reference to the partial derivative.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FormulaTerm">
  <xsd:sequence>
    <xsd:element name="coefficient" type=" xsd:decimal "/>
    <xsd:element name="partialDerivativeReference" type=" PricingParameterDerivativeReference "
    maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GenericDimension

[Table of contents]

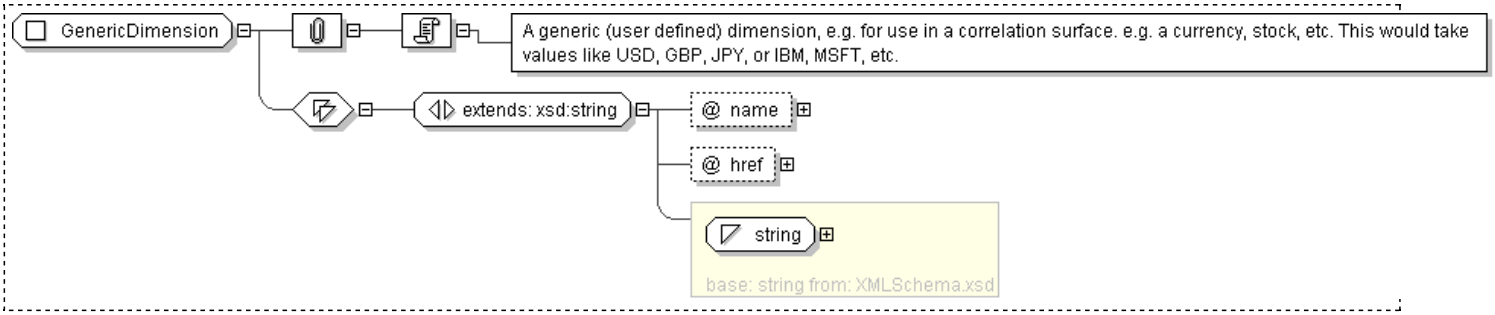
Super-types:	xsd:string < GenericDimension (by extension)
Sub-types:	None

Name	GenericDimension
Used by (from the same schema document)	Model Group PricingStructureIndex.model
Abstract	no
Documentation	A generic (user defined) dimension, e.g. for use in a correlation surface. e.g. a currency, stock, etc. This would take values like USD, GBP, JPY, or IBM, MSFT, etc.

XML Instance Representation

```
<...
 name=" xsd:normalizedString [1]
'The name of the dimension. E.g.: \"Currency\", \"Stock\", \"Issuer\", etc.'

 "
 href=" xsd:IDREF [0..1]
'A reference to an instrument (e.g. currency) that this value represents.'

 ">
   xsd:string
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GenericDimension">
  <xsd:simpleContent>
    <xsd:extension base=" xsd:string ">
      <xsd:attribute name="name" type=" xsd:normalizedString " use="required"/>
      <xsd:attribute name="href" type=" xsd:IDREF " reference="Asset"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InstrumentSet

[Table of contents]

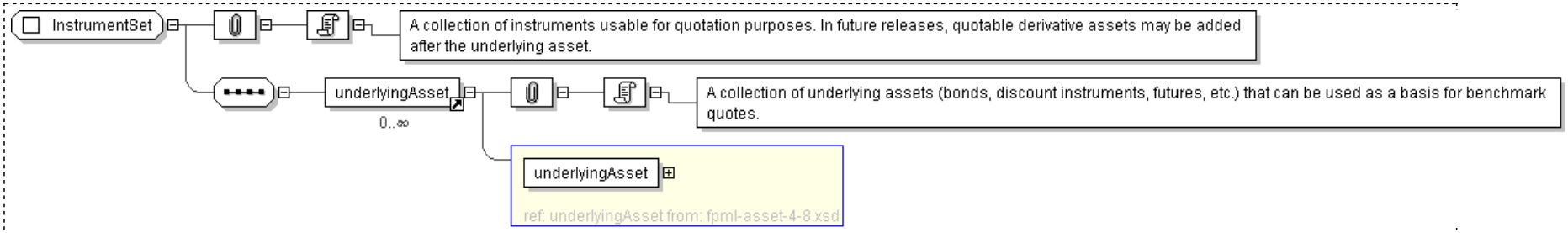
Super-types:	None
Sub-types:	None

Name	InstrumentSet
Used by (from the same schema document)	Complex Type QuotedAssetSet
Abstract	no
Documentation	A collection of instruments usable for quotation purposes. In future releases, quotable derivative assets may be added after the underlying asset.

XML Instance Representation

```
<...>
  <underlyingAsset> ... </underlyingAsset> [0..*]
  'A collection of underlying assets (bonds, discount instruments, futures, etc.) that can be used as a basis for benchmark quotes.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InstrumentSet">
  <xsd:sequence>
    <xsd:element ref=" underlyingAsset " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Market

[Table of contents]

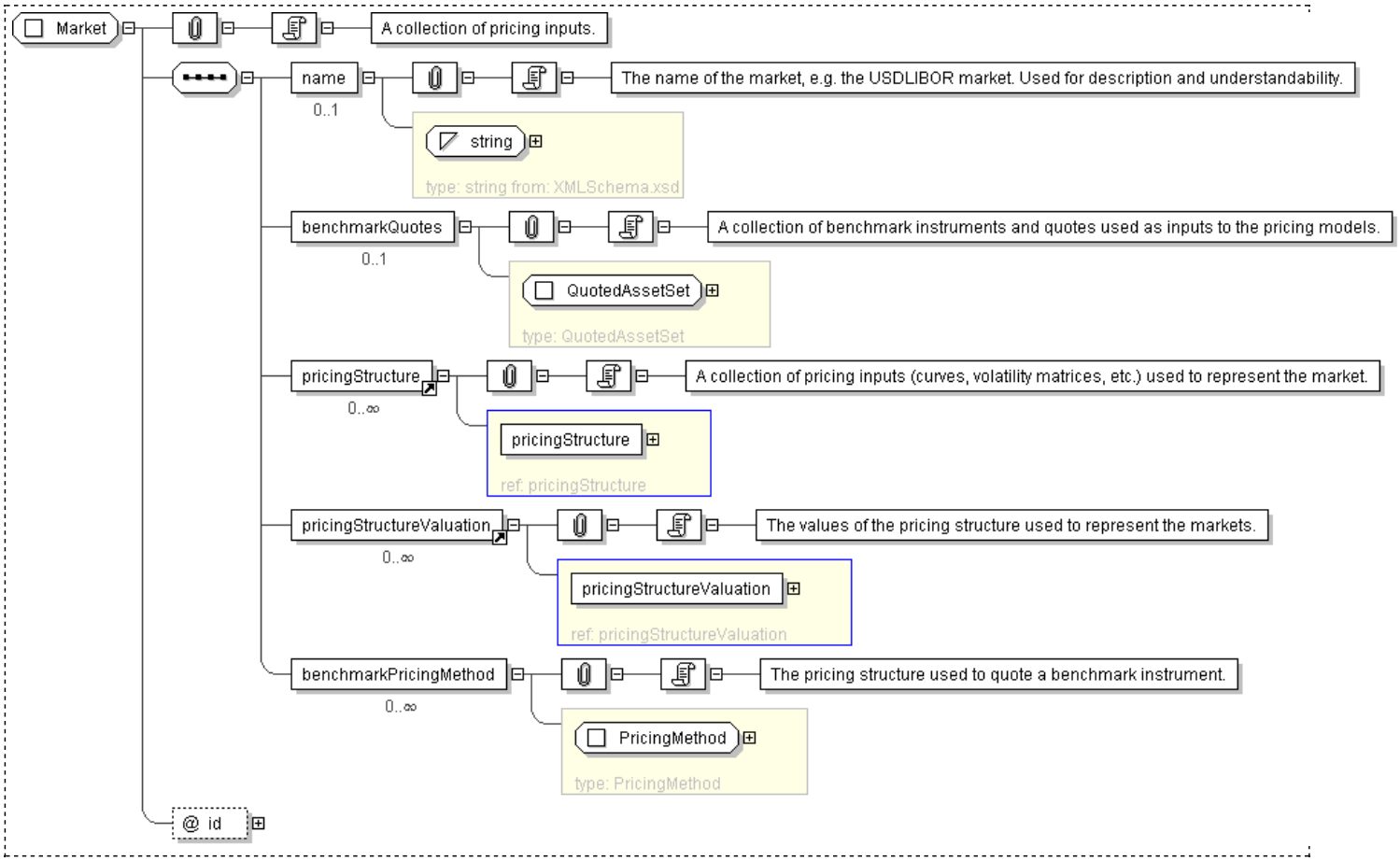
Super-types:	None
Sub-types:	None

Name	Market
Used by (from the same schema document)	Element market
Abstract	no
Documentation	A collection of pricing inputs.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The name of the market, e.g. the USDLIBOR market. Used for description and understandability.'

   <benchmarkQuotes> QuotedAssetSet </benchmarkQuotes> [0..1]
   'A collection of benchmark instruments and quotes used as inputs to the pricing models.'

   <pricingStructure> ... </pricingStructure> [0..*]
   'A collection of pricing inputs (curves, volatility matrices, etc.) used to represent the market.'

   <pricingStructureValuation> ... </pricingStructureValuation> [0..*]
   'The values of the pricing structure used to represent the markets.'

   <benchmarkPricingMethod> PricingMethod </benchmarkPricingMethod> [0..*]
   'The pricing structure used to quote a benchmark instrument.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Market">
```

```
  <xsd:sequence>
    <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="benchmarkQuotes" type=" QuotedAssetSet " minOccurs="0"/>
    <xsd:element ref=" pricingStructure " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref=" pricingStructureValuation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="benchmarkPricingMethod" type=" PricingMethod " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MarketReference

[Table of contents]

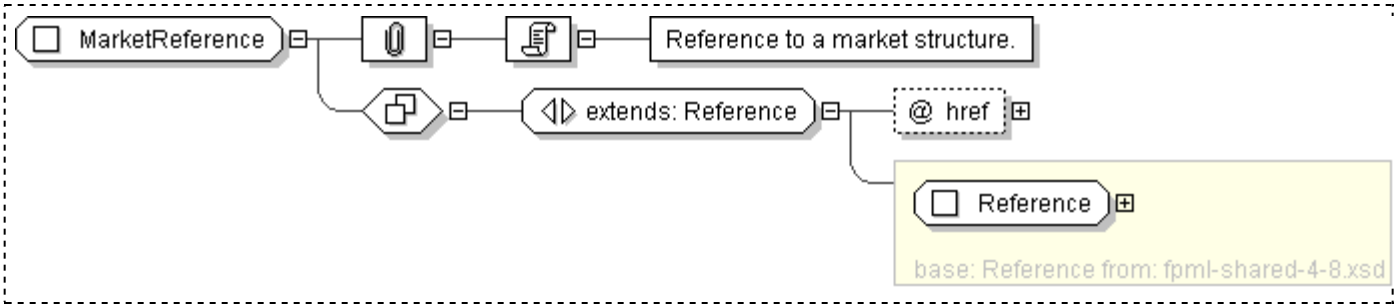
Super-types:	Reference < MarketReference (by extension)
Sub-types:	None

Name	MarketReference
Used by (from the same schema document)	Complex Type ValuationScenario
Abstract	no
Documentation	Reference to a market structure.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MarketReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Market"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PerturbationType

[Table of contents]

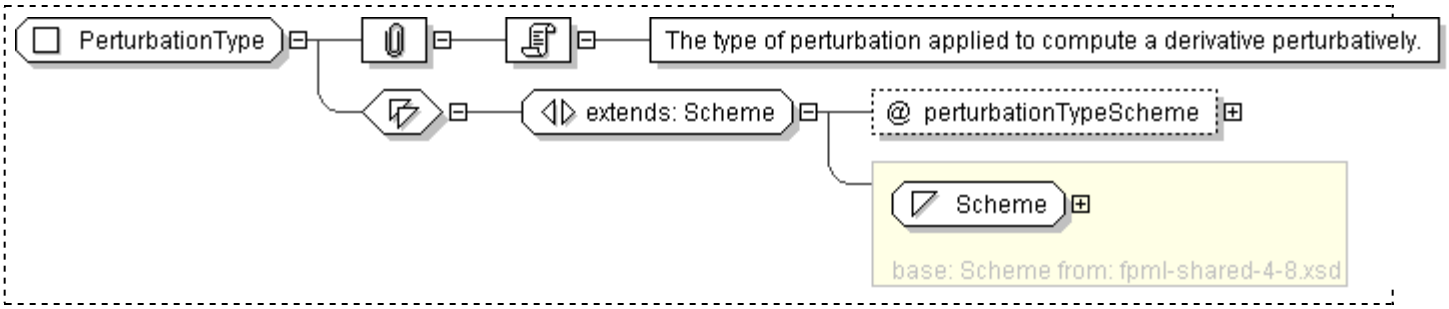
Super-types:	Scheme < PerturbationType (by extension)
Sub-types:	None

Name	PerturbationType
Used by (from the same schema document)	Model Group FiniteDifferenceDerivativeParameters.model
Abstract	no
Documentation	The type of perturbation applied to compute a derivative perturbatively.

XML Instance Representation

```
<...
 perturbationTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PerturbationType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="perturbationTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/perturbation-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionId

[Table of contents]

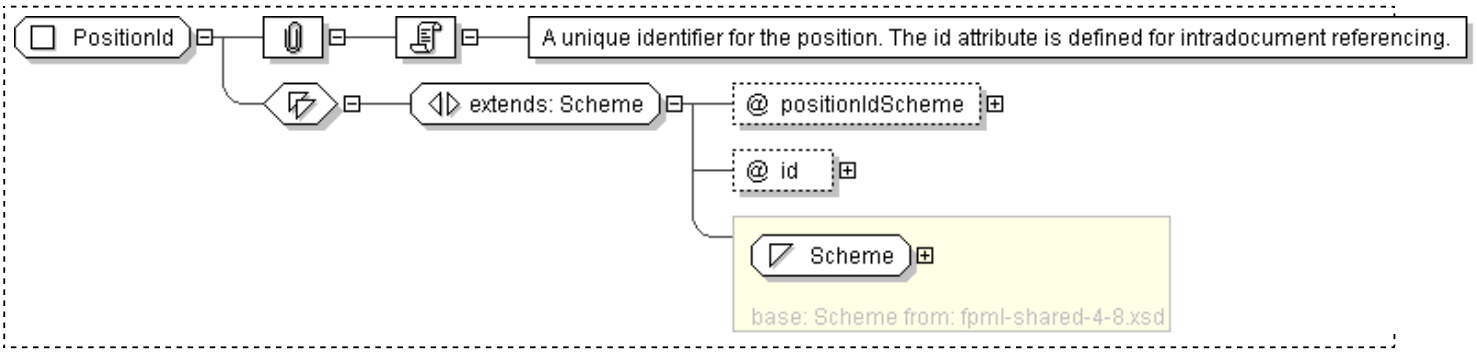
Super-types:	Scheme < PositionId (by extension)
Sub-types:	None

Name	PositionId
Used by (from the same schema document)	Model Group PositionIdAndVersion.model
Abstract	no
Documentation	A unique identifier for the position. The id attribute is defined for intradocument referencing.

XML Instance Representation

```
<...
 positionIdScheme=" xsd:anyURI [0..1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="positionIdScheme" type=" xsd:anyURI "/>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingDataPointCoordinate

[Table of contents]

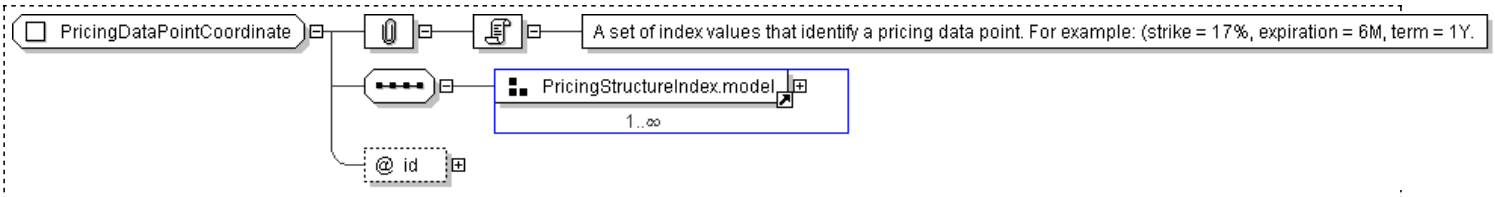
Super-types:	None
Sub-types:	None

Name	PricingDataPointCoordinate
Used by (from the same schema document)	Model Group PricingCoordinateOrReference.model
Abstract	no
Documentation	A set of index values that identify a pricing data point. For example: (strike = 17%, expiration = 6M, term = 1Y.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
  Start Group: PricingStructureIndex.model [1..*]
     Start Choice [1]
        <term> TimeDimension </term> [1]
        'A time dimension that represents the term of a financial instrument, e.g. of a zero-coupon bond on a
        curve, or of an underlying caplet or swap for an option.'

        <expiration> TimeDimension </expiration> [1]
        'A time dimension that represents the time to expiration of an option.'

        <strike> xsd:decimal </strike> [1]
        'A numerical dimension that represents the strike rate or price of an option.'

        <generic> GenericDimension </generic> [1]
     End Choice
  End Group: PricingStructureIndex.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingDataPointCoordinate">
  <xsd:sequence>
    <xsd:group ref=" PricingStructureIndex.model " maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingDataPointCoordinateReference

[Table of contents]

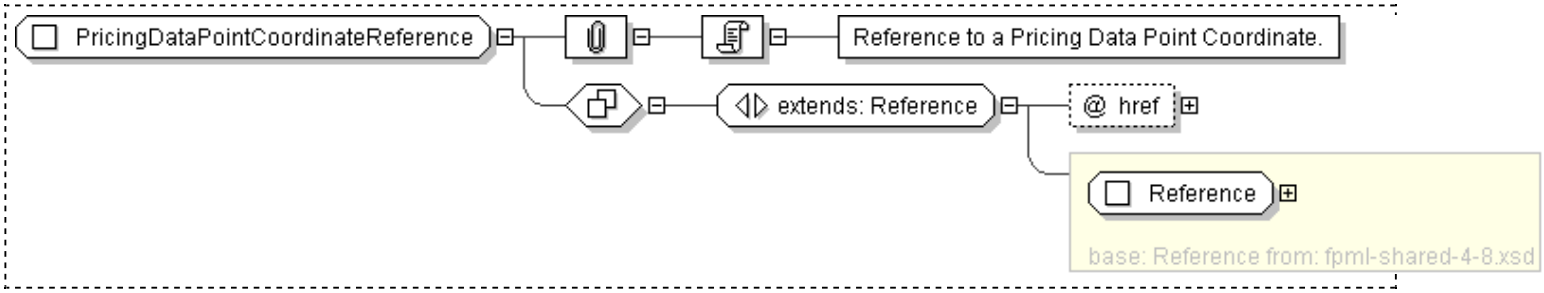
Super-types:	Reference < PricingDataPointCoordinateReference (by extension)
Sub-types:	None

Name	PricingDataPointCoordinateReference
Used by (from the same schema document)	Model Group PricingCoordinateOrReference.model
Abstract	no
Documentation	Reference to a Pricing Data Point Coordinate.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingDataPointCoordinateReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="PricingDataPointCoordinate"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingInputReplacement

[Table of contents]

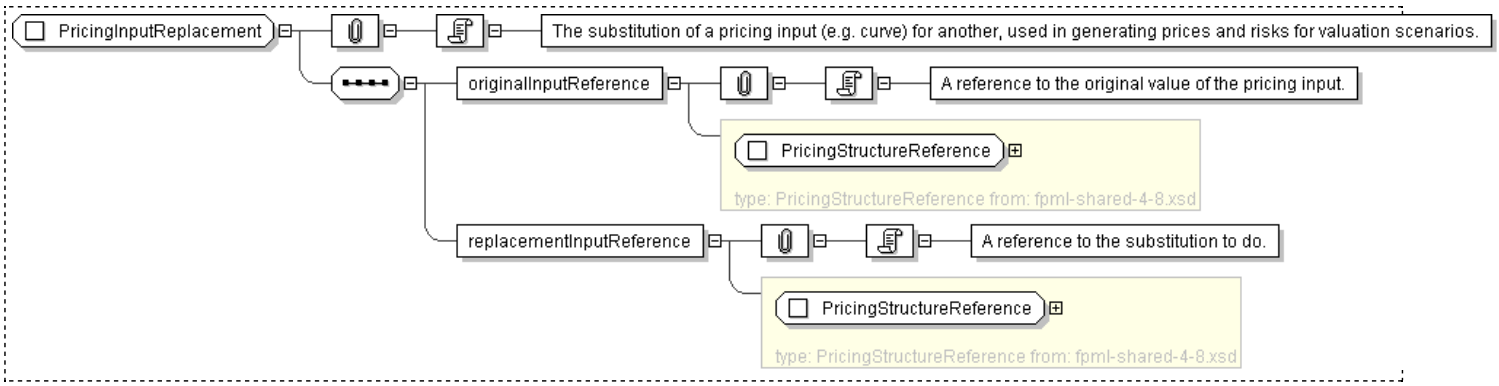
Super-types:	None
Sub-types:	None

Name	PricingInputReplacement
Used by (from the same schema document)	Complex Type ValuationScenario
Abstract	no
Documentation	The substitution of a pricing input (e.g. curve) for another, used in generating prices and risks for valuation scenarios.

XML Instance Representation

```
<...>
  <originalInputReference> PricingStructureReference </originalInputReference> [1]
  'A reference to the original value of the pricing input.'

  <replacementInputReference> PricingStructureReference </replacementInputReference> [1]
  'A reference to the substitution to do.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingInputReplacement">
  <xsd:sequence>
    <xsd:element name="originalInputReference" type=" PricingStructureReference "/>
    <xsd:element name="replacementInputReference" type=" PricingStructureReference "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingInputType

[Table of contents]

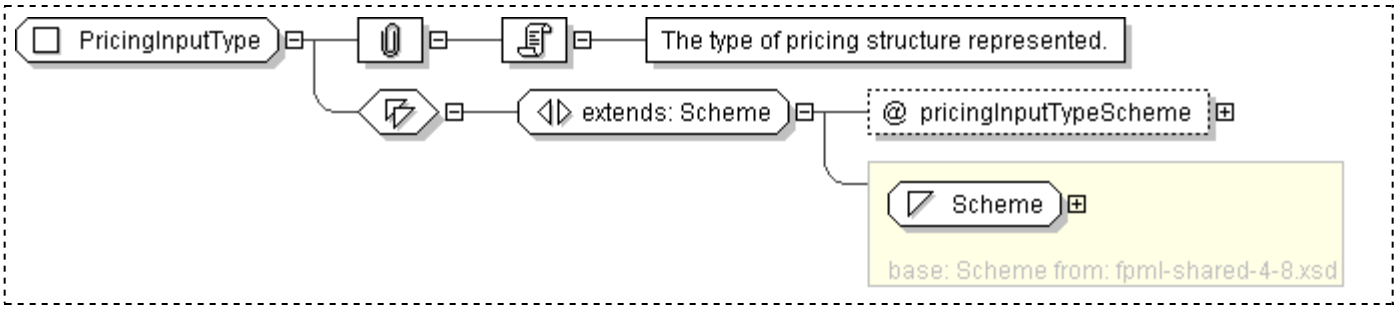
Super-types:	Scheme < PricingInputType (by extension)
Sub-types:	None

Name	PricingInputType
Used by (from the same schema document)	Complex Type SensitivitySetDefinition
Abstract	no
Documentation	The type of pricing structure represented.

XML Instance Representation

```
<...
 pricingInputTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingInputType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="pricingInputTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/pricing-input-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingMethod

[Table of contents]

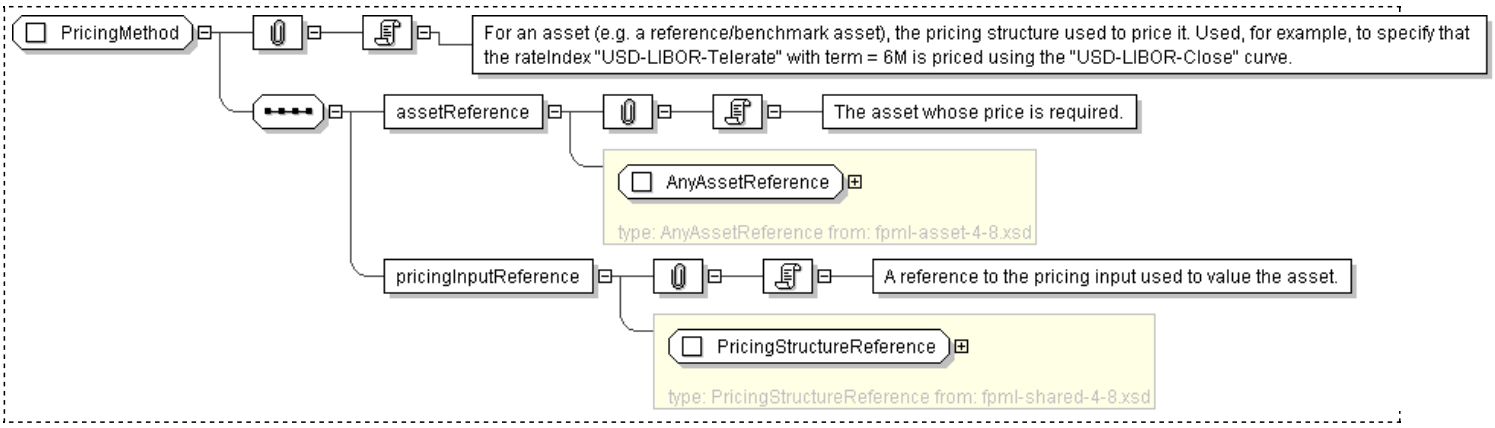
Super-types:	None
Sub-types:	None

Name	PricingMethod
Used by (from the same schema document)	Complex Type Market
Abstract	no
Documentation	For an asset (e.g. a reference/benchmark asset), the pricing structure used to price it. Used, for example, to specify that the rateIndex "USD-LIBOR-Telerate" with term = 6M is priced using the "USD-LIBOR-Close" curve.

XML Instance Representation

```
<...>
  <assetReference> AnyAssetReference </assetReference> [1]
  'The asset whose price is required.'

  <pricingInputReference> PricingStructureReference </pricingInputReference> [1]
  'A reference to the pricing input used to value the asset.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingMethod">
  <xsd:sequence>
    <xsd:element name="assetReference" type=" AnyAssetReference "/>
    <xsd:element name="pricingInputReference" type=" PricingStructureReference "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingParameterDerivative

[Table of contents]

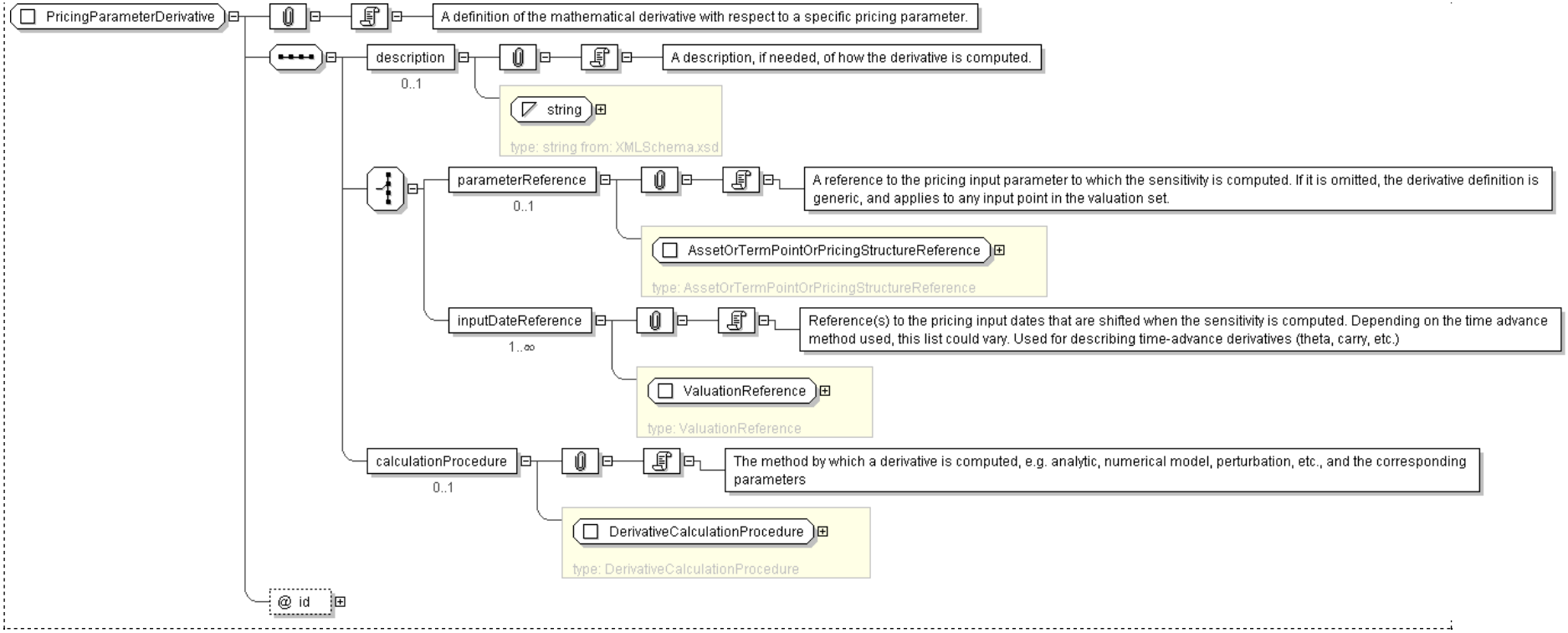
Super-types:	None
Sub-types:	None

Name	PricingParameterDerivative
Used by (from the same schema document)	Model Group ComputedDerivative.model
Abstract	no
Documentation	A definition of the mathematical derivative with respect to a specific pricing parameter.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <description> xsd:string </description> [0..1]
   'A description, if needed, of how the derivative is computed.'

   Start Choice [1]
      <parameterReference> AssetOrTermPointOrPricingStructureReference </parameterReference> [0..1]
      'A reference to the pricing input parameter to which the sensitivity is computed. If it is omitted, the derivative definition is generic, and applies
      to any input point in the valuation set.'

      <inputDateReference> ValuationReference </inputDateReference> [1..*]
      'Reference(s) to the pricing input dates that are shifted when the sensitivity is computed. Depending on the time advance method used, this list could
      vary. Used for describing time-advance derivatives (theta, carry, etc.)'

   End Choice
   <calculationProcedure> DerivativeCalculationProcedure </calculationProcedure> [0..1]
   'The method by which a derivative is computed, e.g. analytic, numerical model, perturbation, etc., and the corresponding parameters'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingParameterDerivative">
   <xsd:sequence>
      <xsd:element name="description" type=" xsd:string " minOccurs="0"/>
      <xsd:choice>
         <xsd:element name="parameterReference" type=" AssetOrTermPointOrPricingStructureReference " minOccurs="0"/>
         <xsd:element name="inputDateReference" type=" ValuationReference " maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:element name="calculationProcedure" type=" DerivativeCalculationProcedure " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingParameterDerivativeReference

[Table of contents]

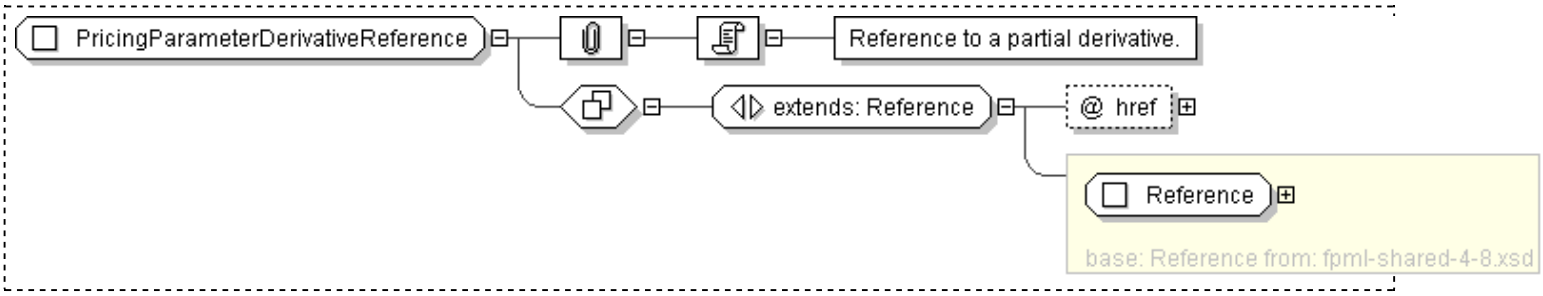
Super-types:	Reference < PricingParameterDerivativeReference (by extension)
Sub-types:	None

Name	PricingParameterDerivativeReference
Used by (from the same schema document)	Complex Type FormulaTerm
Abstract	no
Documentation	Reference to a partial derivative.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingParameterDerivativeReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="PricingParameterDerivative"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingParameterShift

[Table of contents]

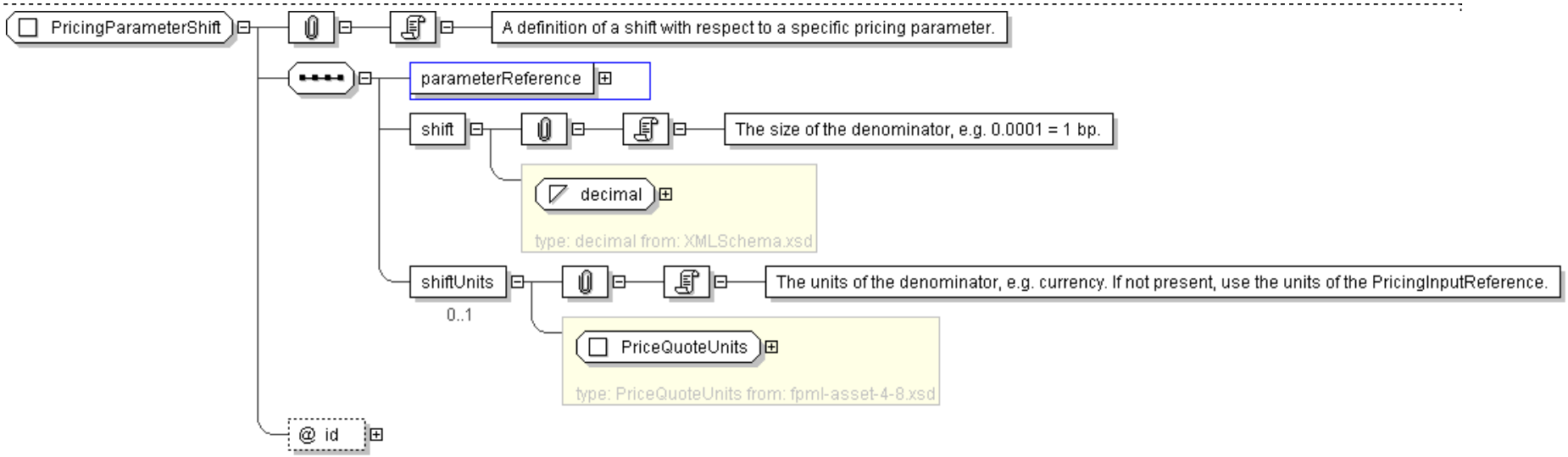
Super-types:	None
Sub-types:	None

Name	PricingParameterShift
Used by (from the same schema document)	Complex Type ValuationScenario
Abstract	no
Documentation	A definition of a shift with respect to a specific pricing parameter.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <parameterReference> AssetOrTermPointOrPricingStructureReference </parameterReference> [1]
   <shift> xsd:decimal </shift> [1]
    'The size of the denominator, e.g. 0.0001 = 1 bp.'

   <shiftUnits> PriceQuoteUnits </shiftUnits> [0..1]
    'The units of the denominator, e.g. currency. If not present, use the units of the PricingInputReference.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingParameterShift">
```

```
  <xsd:sequence>
    <xsd:element name="parameterReference" type=" AssetOrTermPointOrPricingStructureReference "/>
    <xsd:element name="shift" type=" xsd:decimal "/>
    <xsd:element name="shiftUnits" type=" PriceQuoteUnits " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingStructureValuation

[Table of contents]

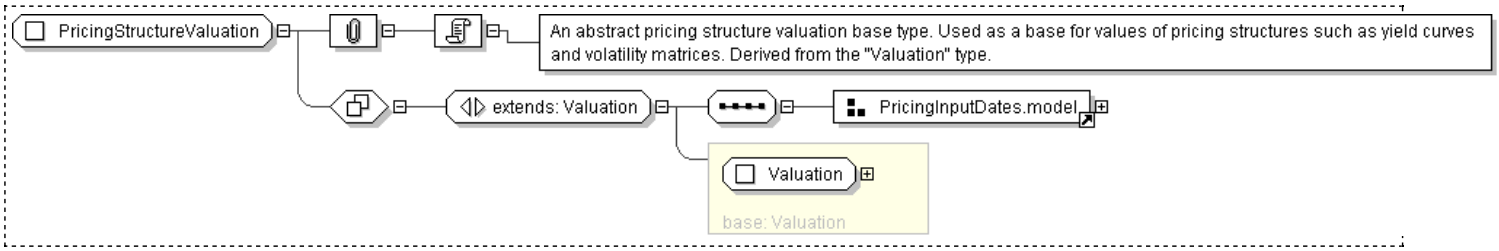
Super-types:	Valuation < PricingStructureValuation (by extension)
Sub-types:	None

Name	PricingStructureValuation
Used by (from the same schema document)	Element pricingStructureValuation
Abstract	no
Documentation	An abstract pricing structure valuation base type. Used as a base for values of pricing structures such as yield curves and volatility matrices. Derived from the "Valuation" type.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a
   ValuationSet, this value is optional and is defaulted from the ValuationSet. If this value occurs in both places,
   the lower level value (i.e. the one here) overrides that in the higher (i.e. ValuationSet).'

   <baseDate> IdentifiedDate </baseDate> [1]
   'The base date for which the structure applies, i.e. the curve date. Normally this will align with the valuation
   date.'

   <spotDate> IdentifiedDate </spotDate> [0..1]
   'The spot settlement date for which the structure applies, normally 0-2 days after the base date. The difference
   between the baseDate and the spotDate is termed the settlement lag, and is sometimes called \"days to spot\".'

   <inputDataDate> IdentifiedDate </inputDataDate> [0..1]
   'The date from which the input data used to construct the pricing input was obtained. Often the same as the
   baseDate, but sometimes the pricing input may be \"rolled forward\", in which input data from one date is used
   to generate a curve for a later date.'

   <endDate> IdentifiedDate </endDate> [0..1]
   'The last date for which data is supplied in this pricing input.'

   <buildDateTime> xsd:dateTime </buildDateTime> [0..1]
   'The date and time when the pricing input was generated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingStructureValuation">
   <xsd:complexContent>
      <xsd:extension base=" Valuation ">
         <xsd:sequence>
            <xsd:group ref=" PricingInputDates.model "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotedAssetSet

[Table of contents]

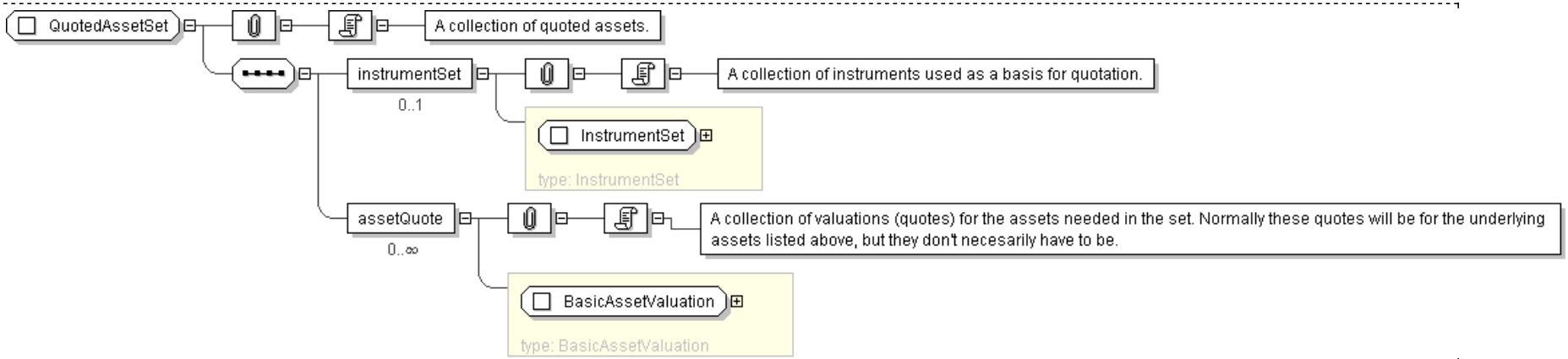
Super-types:	None
Sub-types:	None

Name	QuotedAssetSet
Used by (from the same schema document)	Complex Type Market
Abstract	no
Documentation	A collection of quoted assets.

XML Instance Representation

```
<...>
   <instrumentSet> InstrumentSet </instrumentSet> [0..1]
   'A collection of instruments used as a basis for quotation.'

   <assetQuote> BasicAssetValuation </assetQuote> [0..*]
   'A collection of valuations (quotes) for the assets needed in the set. Normally these quotes will be for the underlying assets
   listed above, but they don\'t necesarily have to be.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotedAssetSet">
   <xsd:sequence>
      <xsd:element name="instrumentSet" type=" InstrumentSet " minOccurs="0"/>
      <xsd:element name="assetQuote" type=" BasicAssetValuation " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SensitivityDefinition

[Table of contents]

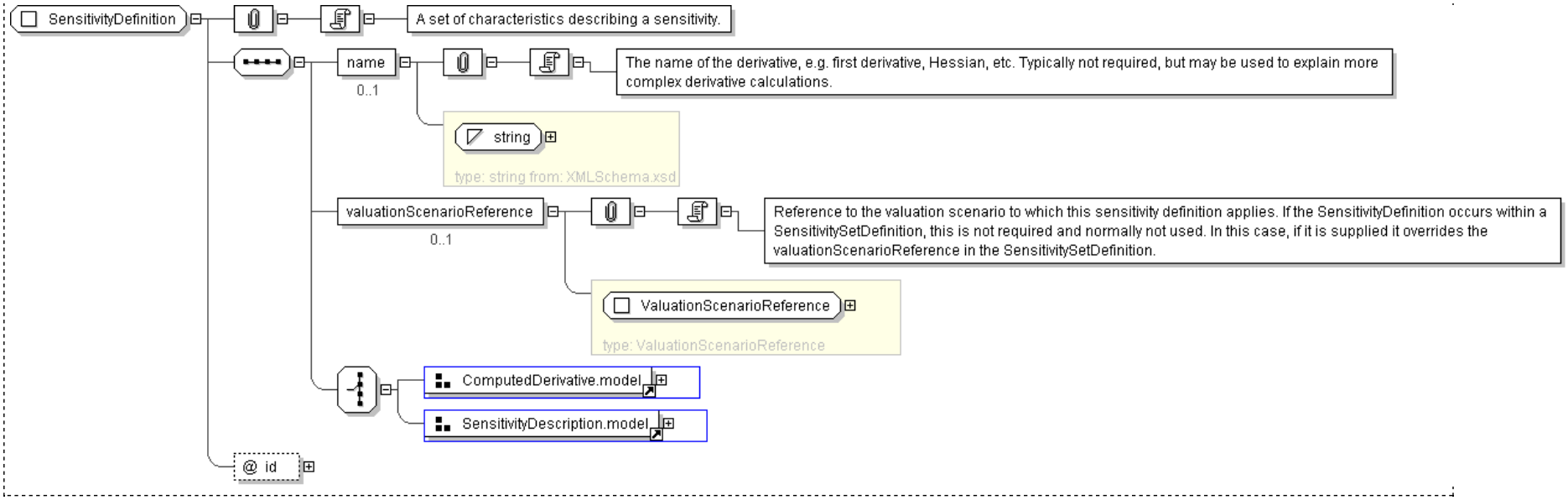
Super-types:	None
Sub-types:	None

Name	SensitivityDefinition
Used by (from the same schema document)	Complex Type SensitivitySetDefinition
Abstract	no
Documentation	A set of characteristics describing a sensitivity.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
    'The name of the derivative, e.g. first derivative, Hessian, etc. Typically not required, but may be used to explain more complex derivative
    calculations.'

    <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
    'Reference to the valuation scenario to which this sensitivity definition applies. If the SensitivityDefinition occurs within a
    SensitivitySetDefinition, this is not required and normally not used. In this case, if it is supplied it overrides the
    valuationScenarioReference in the SensitivitySetDefinition.'

    Start Choice [1]
       <partialDerivative> PricingParameterDerivative </partialDerivative> [1..*]
        'A partial derivative of the measure with respect to an input.'

       <formula> DerivativeFormula </formula> [0..1]
        'A formula defining how to compute the derivative from the partial derivatives. If absent, the derivative is just the product of the partial
        derivatives. Normally only required for more higher-order derivatives, e.g. Hessians.'

       Start Choice [1]
          <term> TimeDimension </term> [1]
           'The time dimension of the sensitivity point (tenor and/or date).'

          Start Group: PricingCoordinateOrReference.model [1..*]
'The input coordinates, or references to them (e.g. expiration, strike, tenor).'


             Start Choice [1]
                <coordinate> PricingDataPointCoordinate </coordinate> [1]
                 'An explicit, filled in data point coordinate. This might specify expiration, strike, etc.'

                <coordinateReference> PricingDataPointCoordinateReference </coordinateReference> [1]
                 'A reference to a pricing data point coordinate within this document.'

             End Choice
          End Group: PricingCoordinateOrReference.model
       End Choice
    End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SensitivityDefinition">
  <xsd:sequence>
    <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="valuationScenarioReference" type=" ValuationScenarioReference " minOccurs="0"/>
    <xsd:choice>
      <xsd:group ref=" ComputedDerivative.model "/>
      <xsd:group ref=" SensitivityDescription.model "/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SensitivitySetDefinition

[Table of contents]

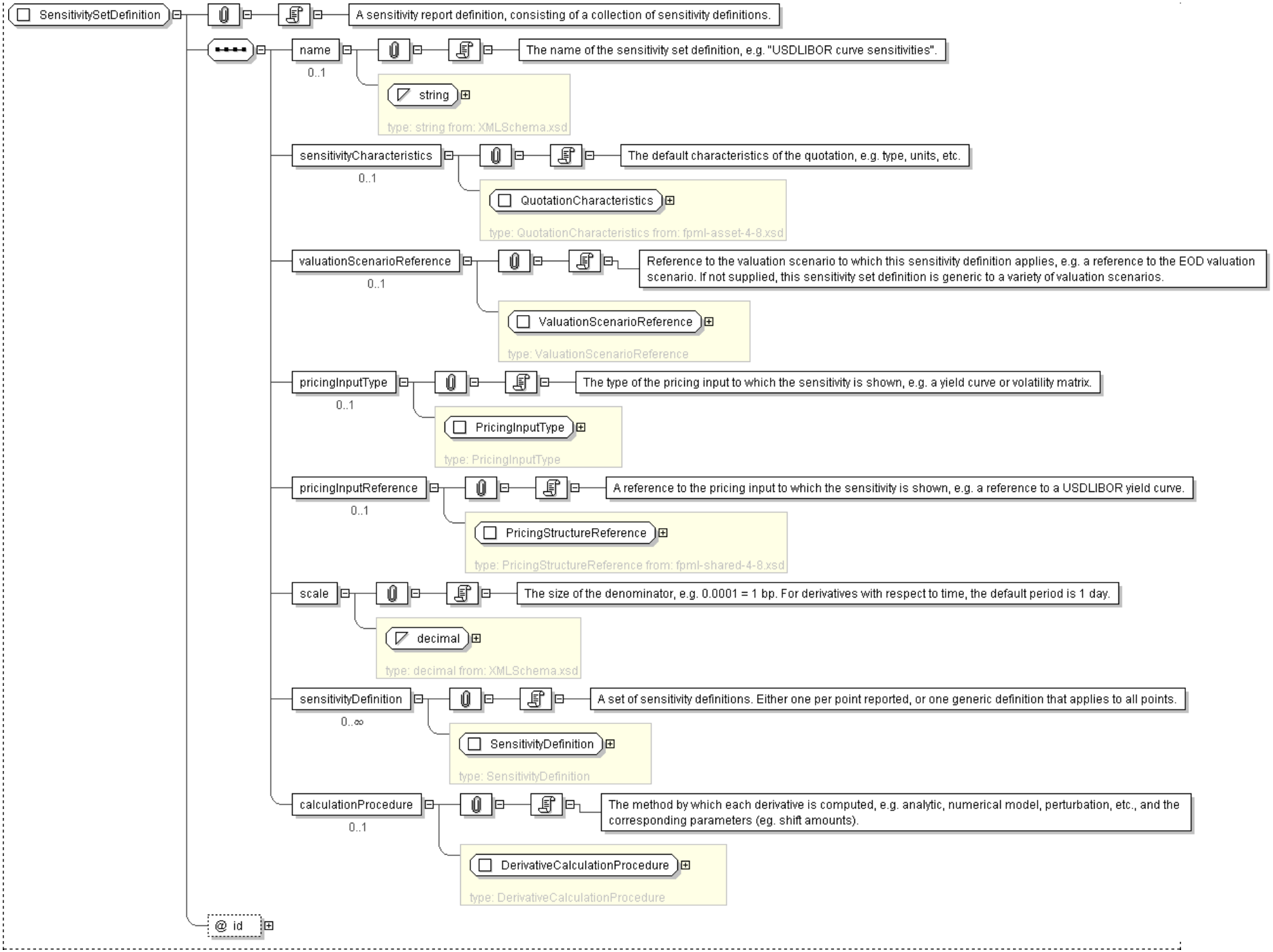
Super-types:	None
Sub-types:	None

Name	SensitivitySetDefinition
Abstract	no
Documentation	A sensitivity report definition, consisting of a collection of sensitivity definitions.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The name of the sensitivity set definition, e.g. \"USDLIBOR curve sensitivities\".'

   <sensitivityCharacteristics> QuotationCharacteristics </sensitivityCharacteristics> [0..1]
   'The default characteristics of the quotation, e.g. type, units, etc.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'Reference to the valuation scenario to which this sensitivity definition applies, e.g. a reference to the EOD valuation scenario. If not
   supplied, this sensitivity set definition is generic to a variety of valuation scenarios.'

   <pricingInputType> PricingInputType </pricingInputType> [0..1]
   'The type of the pricing input to which the sensitivity is shown, e.g. a yield curve or volatility matrix.'

   <pricingInputReference> PricingStructureReference </pricingInputReference> [0..1]
   'A reference to the pricing input to which the sensitivity is shown, e.g. a reference to a USDLIBOR yield curve.'

   <scale> xsd:decimal </scale> [1]
   'The size of the denominator, e.g. 0.0001 = 1 bp. For derivatives with respect to time, the default period is 1 day.'

   <sensitivityDefinition> SensitivityDefinition </sensitivityDefinition> [0..*]
   'A set of sensitivity definitions. Either one per point reported, or one generic definition that applies to all points.'

   <calculationProcedure> DerivativeCalculationProcedure </calculationProcedure> [0..1]
   'The method by which each derivative is computed, e.g. analytic, numerical model, perturbation, etc., and the corresponding parameters (eg.
   shift amounts).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SensitivitySetDefinition">
  <xsd:sequence>
    <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="sensitivityCharacteristics" type=" QuotationCharacteristics " minOccurs="0"/>
```

```
      <xsd:element name="valuationScenarioReference" type=" ValuationScenarioReference " minOccurs="0"/>
      <xsd:element name="pricingInputType" type=" PricingInputType " minOccurs="0"/>
      <xsd:element name="pricingInputReference" type=" PricingStructureReference " minOccurs="0"/>
      <xsd:element name="scale" type=" xsd:decimal "/>
      <xsd:element name="sensitivityDefinition" type=" SensitivityDefinition " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="calculationProcedure" type=" DerivativeCalculationProcedure " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SensitivitySetDefinitionReference

[Table of contents]

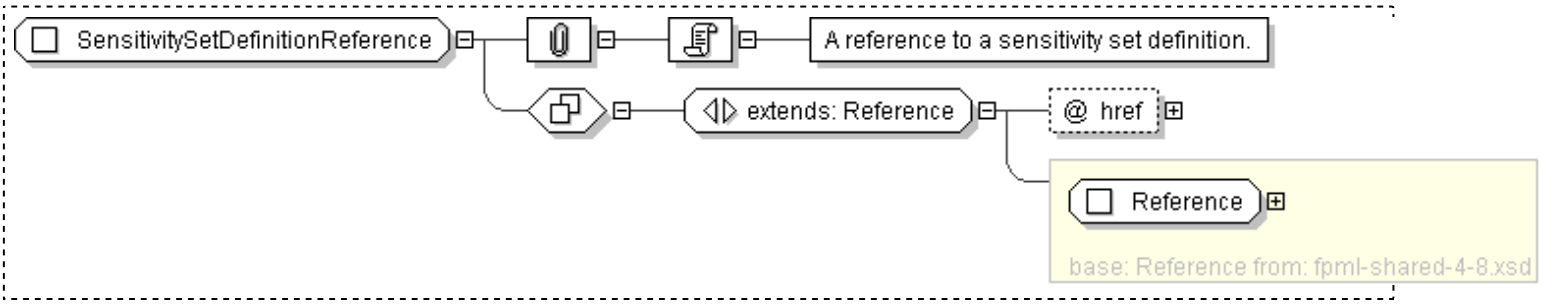
Super-types:	Reference < SensitivitySetDefinitionReference (by extension)
Sub-types:	None

Name	SensitivitySetDefinitionReference
Abstract	no
Documentation	A reference to a sensitivity set definition.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SensitivitySetDefinitionReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="SensitivitySetDefinition"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TimeDimension

[Table of contents]

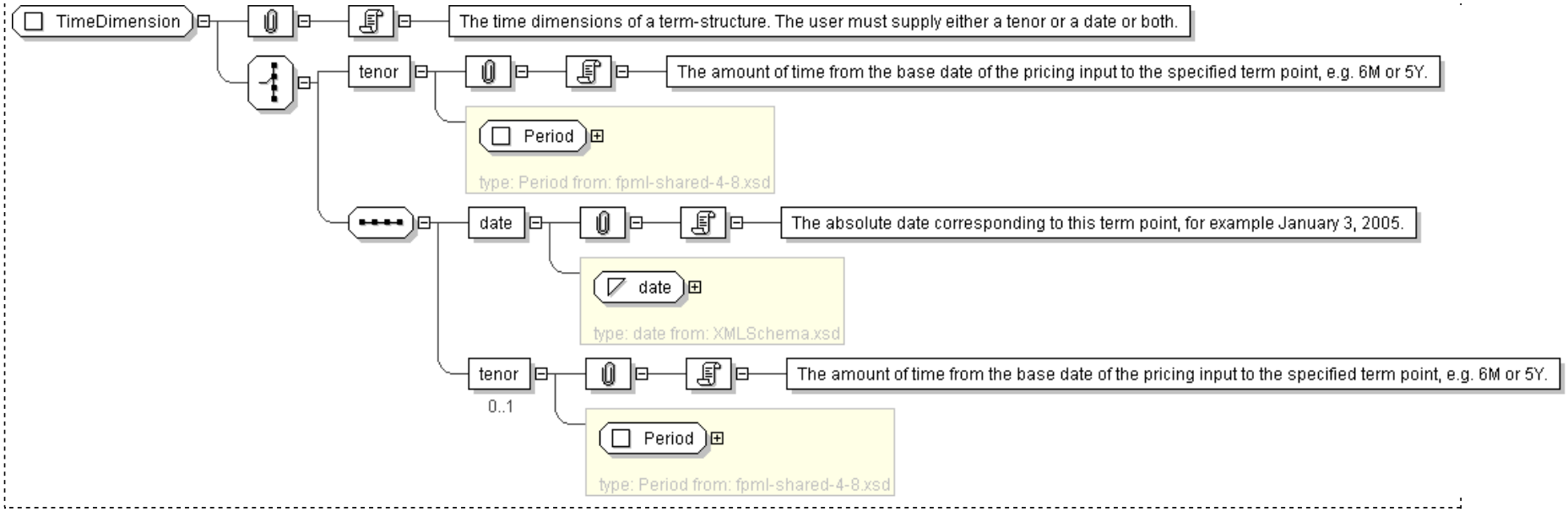
Super-types:	None
Sub-types:	None

Name	TimeDimension
Used by (from the same schema document)	Model Group PricingStructureIndex.model , Model Group PricingStructureIndex.model , Model Group SensitivityDescription.model
Abstract	no
Documentation	The time dimensions of a term-structure. The user must supply either a tenor or a date or both.

XML Instance Representation

```
<...>
   Start Choice [1]
      <tenor> Period </tenor> [1]
      'The amount of time from the base date of the pricing input to the specified term point, e.g. 6M or 5Y.'

      <date> xsd:date </date> [1]
      'The absolute date corresponding to this term point, for example January 3, 2005.'

      <tenor> Period </tenor> [0..1]
      'The amount of time from the base date of the pricing input to the specified term point, e.g. 6M or 5Y.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TimeDimension">
  <xsd:choice>
    <xsd:element name="tenor" type=" Period "/>
    <xsd:sequence>
      <xsd:element name="date" type=" xsd:date "/>
      <xsd:element name="tenor" type=" Period " minOccurs="0"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Valuation

[Table of contents]

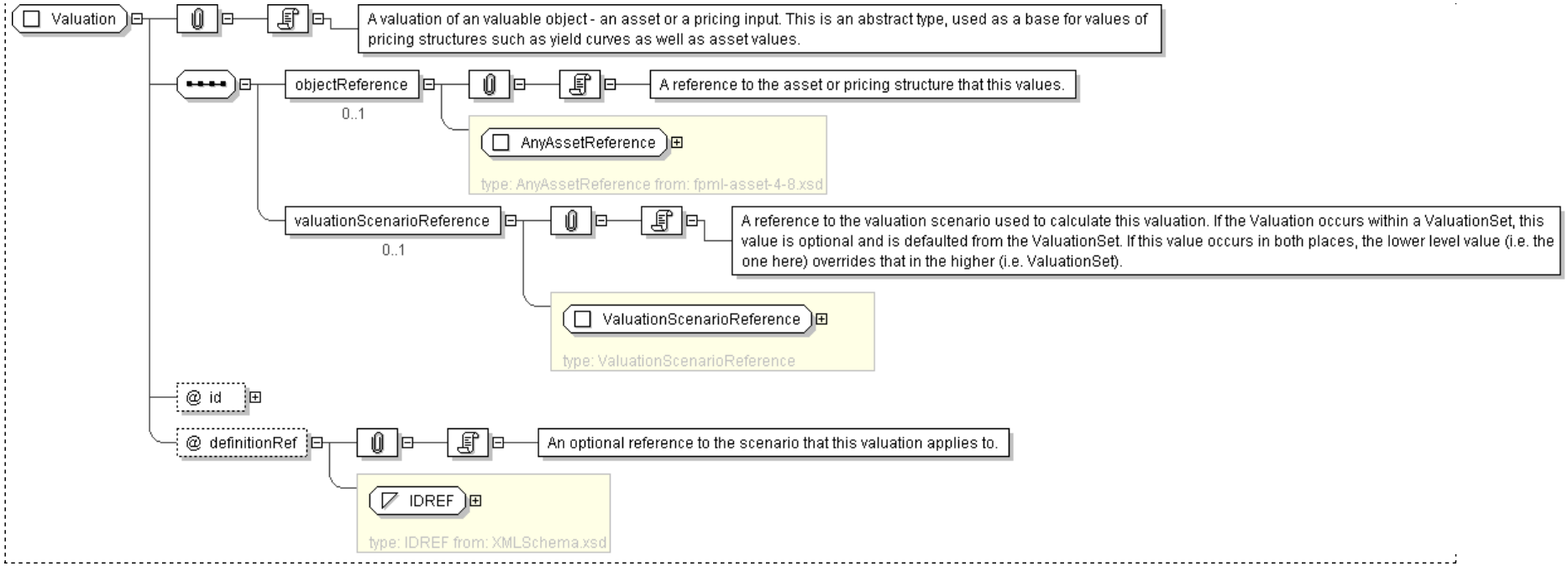
Super-types:	None
Sub-types:	• BasicAssetValuation (by extension) • PricingStructureValuation (by extension)

Name	Valuation
Abstract	no
Documentation	A valuation of an valuable object - an asset or a pricing input. This is an abstract type, used as a base for values of pricing structures such as yield curves as well as asset values.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation occurs within a ValuationSet, this value is
   optional and is defaulted from the ValuationSet. If this value occurs in both places, the lower level value (i.e. the one here) overrides
   that in the higher (i.e. ValuationSet).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Valuation">
  <xsd:sequence>
    <xsd:element name="objectReference" type=" AnyAssetReference " minOccurs="0"/>
    <xsd:element name="valuationScenarioReference" type=" ValuationScenarioReference " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
  <xsd:attribute name="definitionRef" type=" xsd:IDREF " reference="ValuationScenario"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationReference

[Table of contents]

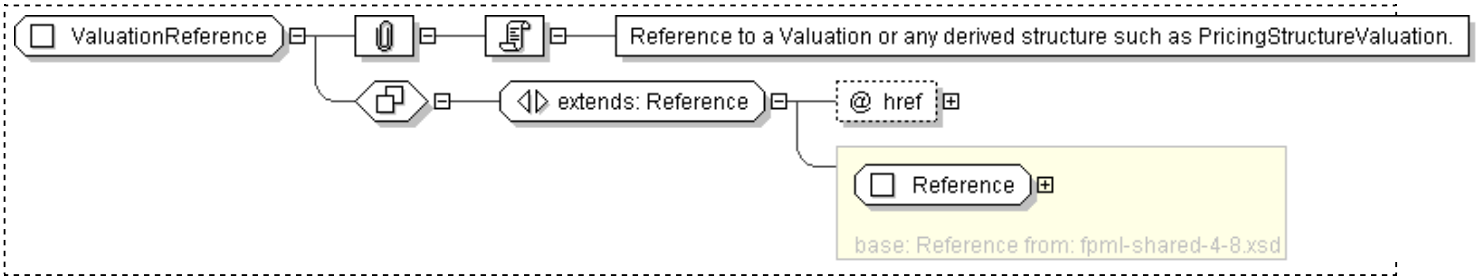
Super-types:	Reference < ValuationReference (by extension)
Sub-types:	None

Name	ValuationReference
Used by (from the same schema document)	Complex Type PricingParameterDerivative
Abstract	no
Documentation	Reference to a Valuation or any derived structure such as PricingStructureValuation.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="Valuation"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationScenario

[Table of contents]

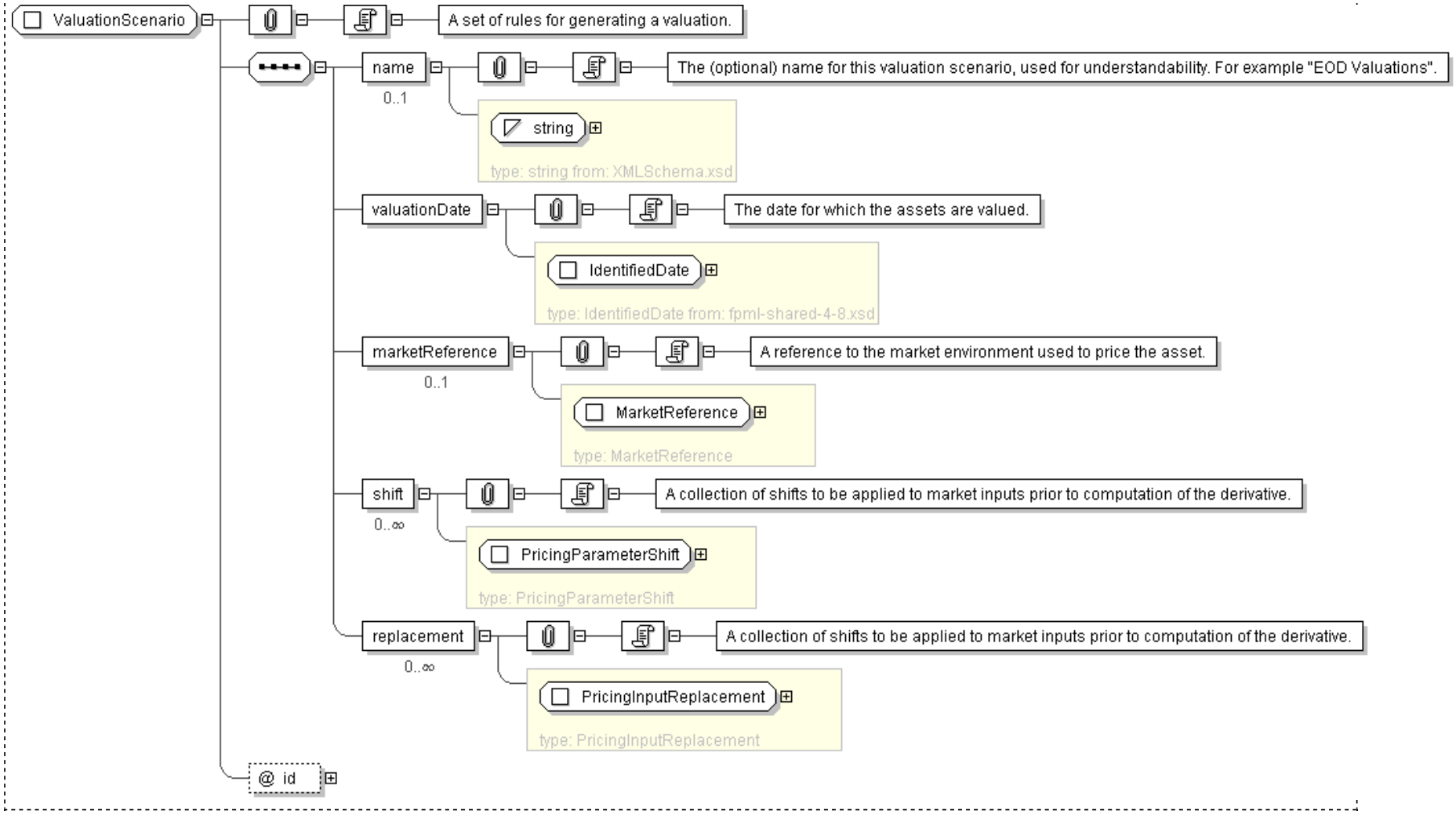
Super-types:	None
Sub-types:	None

Name	ValuationScenario
Abstract	no
Documentation	A set of rules for generating a valuation.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The (optional) name for this valuation scenario, used for understandability. For example \"EOD Valuations\".'

   <valuationDate> IdentifiedDate </valuationDate> [1]
   'The date for which the assets are valued.'

   <marketReference> MarketReference </marketReference> [0..1]
   'A reference to the market environment used to price the asset.'

   <shift> PricingParameterShift </shift> [0..*]
   'A collection of shifts to be applied to market inputs prior to computation of the derivative.'

   <replacement> PricingInputReplacement </replacement> [0..*]
   'A collection of shifts to be applied to market inputs prior to computation of the derivative.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationScenario">
  <xsd:sequence>
    <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="valuationDate" type=" IdentifiedDate "/>
    <xsd:element name="marketReference" type=" MarketReference " minOccurs="0"/>
    <xsd:element name="shift" type=" PricingParameterShift " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="replacement" type=" PricingInputReplacement " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationScenarioReference

[Table of contents]

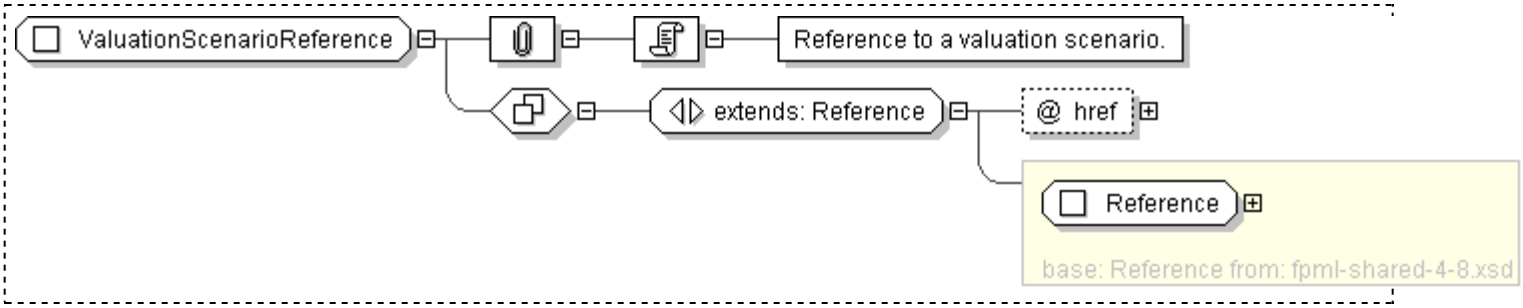
Super-types:	Reference < ValuationScenarioReference (by extension)
Sub-types:	None

Name	ValuationScenarioReference
Used by (from the same schema document)	Complex Type SensitivityDefinition , Complex Type SensitivitySetDefinition , Complex Type Valuation
Abstract	no
Documentation	Reference to a valuation scenario.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationScenarioReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="ValuationScenario"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: WeightedPartialDerivative

[Table of contents]

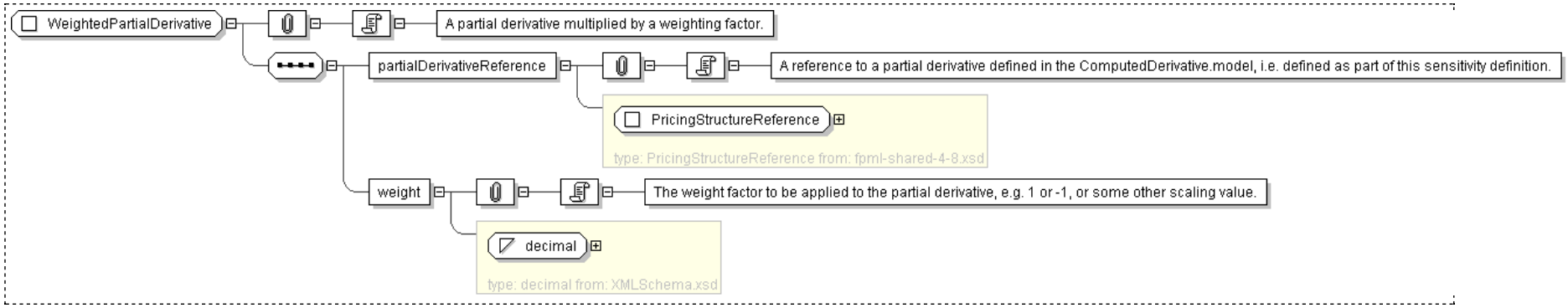
Super-types:	None
Sub-types:	None

Name	WeightedPartialDerivative
Used by (from the same schema document)	Complex Type DenominatorTerm
Abstract	no
Documentation	A partial derivative multiplied by a weighting factor.

XML Instance Representation

```
<...>
   <partialDerivativeReference> PricingStructureReference </partialDerivativeReference> [1]
   'A reference to a partial derivative defined in the ComputedDerivative.model, i.e. defined as part of this sensitivity definition.'

   <weight> xsd:decimal </weight> [1]
   'The weight factor to be applied to the partial derivative, e.g. 1 or -1, or some other scaling value.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="WeightedPartialDerivative">
   <xsd:sequence>
      <xsd:element name="partialDerivativeReference" type=" PricingStructureReference "/>
      <xsd:element name="weight" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: americanExercise
  - Element: bermudaExercise
  - Element: europeanExercise
  - Element: exercise
  - Element: product
- Global Definitions
  - Complex Type: Account
  - Complex Type: AccountId
  - Complex Type: AccountReference
  - Complex Type: Address
  - Complex Type: AdjustableDate
  - Complex Type: AdjustableDate2
  - Complex Type: AdjustableDates
  - Complex Type: AdjustableDatesOrRelativeDateOffset
  - Complex Type: AdjustableOrRelativeAndAdjustedDate
  - Complex Type: AdjustableOrRelativeDate
  - Complex Type: AdjustableOrRelativeDates
  - Complex Type: AdjustableRelativeOrPeriodicDates
  - Complex Type: AdjustableRelativeOrPeriodicDates2
  - Complex Type: AdjustedRelativeDateOffset
  - Complex Type: AmericanExercise
  - Complex Type: AmountReference
  - Complex Type: AmountSchedule
  - Complex Type: AutomaticExercise
  - Complex Type: AverageDailyTradingVolumeLimit
  - Complex Type: Beneficiary
  - Complex Type: BermudaExercise
  - Complex Type: BrokerConfirmation
  - Complex Type: BrokerConfirmationType
  - Complex Type: BusinessCenter
  - Complex Type: BusinessCenterTime
  - Complex Type: BusinessCenters
  - Complex Type: BusinessCentersReference
  - Complex Type: BusinessDateRange
  - Complex Type: BusinessDayAdjustments
  - Complex Type: BusinessDayAdjustmentsReference
  - Complex Type: CalculationAgent
  - Complex Type: CalculationPeriodFrequency
  - Complex Type: CashSettlementReferenceBanks
  - Complex Type: CashflowType
  - Complex Type: ClearanceSystem
  - Complex Type: ContractualDefinitions
  - Complex Type: ContractualMatrix
  - Complex Type: ContractualSupplement
  - Complex Type: ContractualTermsSupplement
  - Complex Type: CorrespondentInformation
  - Complex Type: Country
  - Complex Type: CreditSeniority
  - Complex Type: CreditSupportAgreement
  - Complex Type: CreditSupportAgreementType
  - Complex Type: Currency
  - Complex Type: DateList

- Complex Type: DateOffset
- Complex Type: DateRange
- Complex Type: DateReference
- Complex Type: DateTimeList
- Complex Type: DayCountFraction
- Complex Type: DeterminationMethod
- Complex Type: DeterminationMethodReference
- Complex Type: Documentation
- Complex Type: Empty
- Complex Type: EntityId
- Complex Type: EntityName
- Complex Type: EuropeanExercise
- Complex Type: ExchangeId
- Complex Type: Exercise
- Complex Type: ExerciseFee
- Complex Type: ExerciseFeeSchedule
- Complex Type: ExerciseNotice
- Complex Type: ExerciseProcedure
- Complex Type: FloatingRate
- Complex Type: FloatingRateCalculation
- Complex Type: FloatingRateIndex
- Complex Type: ForecastRateIndex
- Complex Type: Formula
- Complex Type: FormulaComponent
- Complex Type: Frequency
- Complex Type: FutureValueAmount
- Complex Type: FxCashSettlement
- Complex Type: FxFixing
- Complex Type: FxRate
- Complex Type: FxSpotRateSource
- Complex Type: GoverningLaw
- Complex Type: IdentifiedCurrency
- Complex Type: IdentifiedCurrencyReference
- Complex Type: IdentifiedDate
- Complex Type: IdentifiedPayerReceiver
- Complex Type: InformationProvider
- Complex Type: InformationSource
- Complex Type: InstrumentId
- Complex Type: InterestAccrualsCompoundingMethod
- Complex Type: InterestAccrualsMethod
- Complex Type: IntermediaryInformation
- Complex Type: InterpolationMethod
- Complex Type: Leg
- Complex Type: LegalEntity
- Complex Type: LegalEntityReference
- Complex Type: MainPublication
- Complex Type: ManualExercise
- Complex Type: MasterAgreement
- Complex Type: MasterAgreementType
- Complex Type: MasterAgreementVersion
- Complex Type: MasterConfirmation
- Complex Type: MasterConfirmationAnnexType
- Complex Type: MasterConfirmationType
- Complex Type: Math
- Complex Type: MatrixTerm
- Complex Type: MatrixType
- Complex Type: MimeType
- Complex Type: Money
- Complex Type: MoneyBase
- Complex Type: MultipleExercise

Complex Type: NonNegativeAmountSchedule
- Complex Type: NonNegativeMoney
- Complex Type: NonNegativePayment
- Complex Type: NonNegativeSchedule
- Complex Type: NonNegativeStep
- Complex Type: NotionalAmount
- Complex Type: NotionalAmountReference
- Complex Type: NotionalReference
- Complex Type: Offset
- Complex Type: OffsetPrevailingTime
- Complex Type: PartialExercise
- Complex Type: Party
- Complex Type: PartyId
- Complex Type: PartyOrAccountReference
- Complex Type: PartyOrTradeSideReference
- Complex Type: PartyReference
- Complex Type: Payment
- Complex Type: PaymentBase
- Complex Type: PaymentBaseExtended
- Complex Type: PaymentCurrency
- Complex Type: PaymentReference
- Complex Type: PaymentType
- Complex Type: Period
- Complex Type: PeriodicDates
- Complex Type: PositiveAmountSchedule
- Complex Type: PositiveMoney
- Complex Type: PositivePayment
- Complex Type: PositiveSchedule
- Complex Type: PositiveStep
- Complex Type: PrevailingTime
- Complex Type: PricingStructure
- Complex Type: PricingStructureReference
- Complex Type: PrincipalExchanges
- Complex Type: Product
- Complex Type: ProductId
- Complex Type: ProductReference
- Complex Type: ProductType
- Complex Type: QuotedCurrencyPair
- Complex Type: Rate
- Complex Type: RateObservation
- Complex Type: RateReference
- Complex Type: RateSourcePage
- Complex Type: Reference
- Complex Type: ReferenceAmount
- Complex Type: ReferenceBank
- Complex Type: ReferenceBankId
- Complex Type: RelativeDateOffset
- Complex Type: RelativeDateSequence
- Complex Type: RelativeDates
- Complex Type: RequiredIdentifierDate
- Complex Type: ResetFrequency
- Complex Type: ReturnSwapNotionalAmountReference
- Complex Type: Rounding
- Complex Type: Routing
- Complex Type: RoutingExplicitDetails
- Complex Type: RoutingId
- Complex Type: RoutingIds
- Complex Type: RoutingIdsAndExplicitDetails
- Complex Type: Schedule
- Complex Type: ScheduleReference

- Complex Type: SettlementInformation
- Complex Type: SettlementInstruction
- Complex Type: SettlementMethod
- Complex Type: SettlementPriceDefaultElection
- Complex Type: SettlementPriceSource
- Complex Type: SettlementRateSource
- Complex Type: SharedAmericanExercise
- Complex Type: SimplePayment
- Complex Type: SplitSettlement
- Complex Type: SpreadSchedule
- Complex Type: SpreadScheduleReference
- Complex Type: SpreadScheduleType
- Complex Type: Step
- Complex Type: StepBase
- Complex Type: StreetAddress
- Complex Type: Strike
- Complex Type: StrikeSchedule
- Complex Type: Stub
- Complex Type: StubValue
- Complex Type: TimezoneLocation
- Model Group: BusinessCentersOrReference.model
- Model Group: BuyerSeller.model
- Model Group: FloatingRateIndex.model
- Model Group: PartialExercise.model
- Model Group: PayerReceiver.model
- Model Group: PaymentDiscounting.model
- Model Group: Period.model
- Model Group: Premium.model
- Model Group: Product.model
- Model Group: RoutingExplicitDetails.model
- Model Group: RoutingIdentification.model
- Model Group: SettlementAmountOrCurrency.model
- Model Group: VersionHistory.model
- Simple Type: CorrelationValue
- Simple Type: HourMinuteTime
- Simple Type: NonNegativeDecimal
- Simple Type: PositiveDecimal
- Simple Type: RestrictedPercentage
- Simple Type: Scheme
- Simple Type: Token60

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-enum-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-enum-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].

- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in `bold`.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
    - If the element/attribute's type is in the schema, a link is provided to it.
    - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
  <complexContent>
    <extension base=" Address ">
      <sequence>
        <element name="state" type=" AusStates "/>
        <element name="postcode">
          <simpleType>
            <restriction base=" string ">
              <pattern value="[1-9][0-9]{3}"/>
            </restriction>
          </simpleType>
        </element>
      </sequence>
      <attribute name="country" type=" string " fixed="Australia"/>
    </extension>
  </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

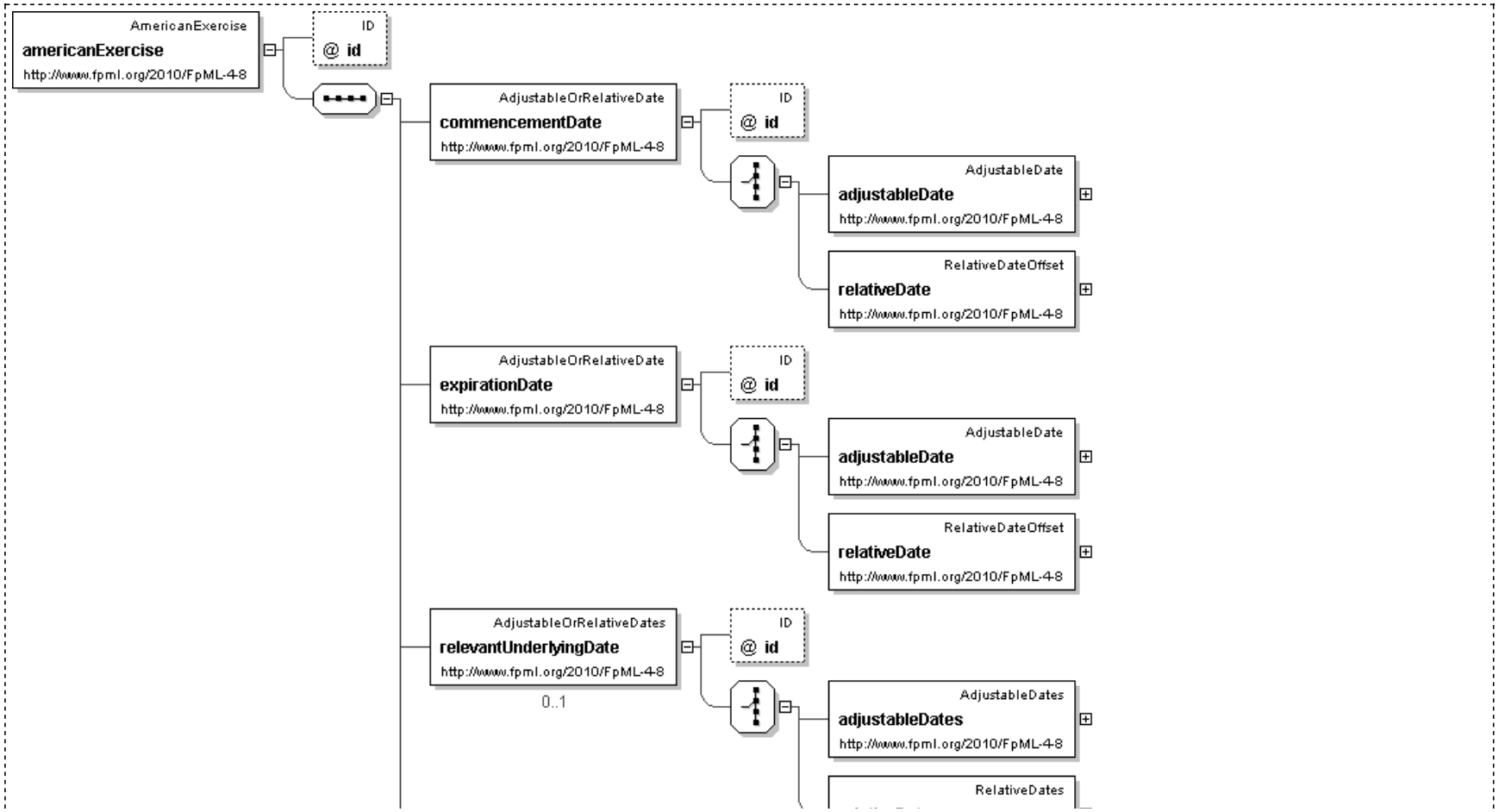
Element: americanExercise

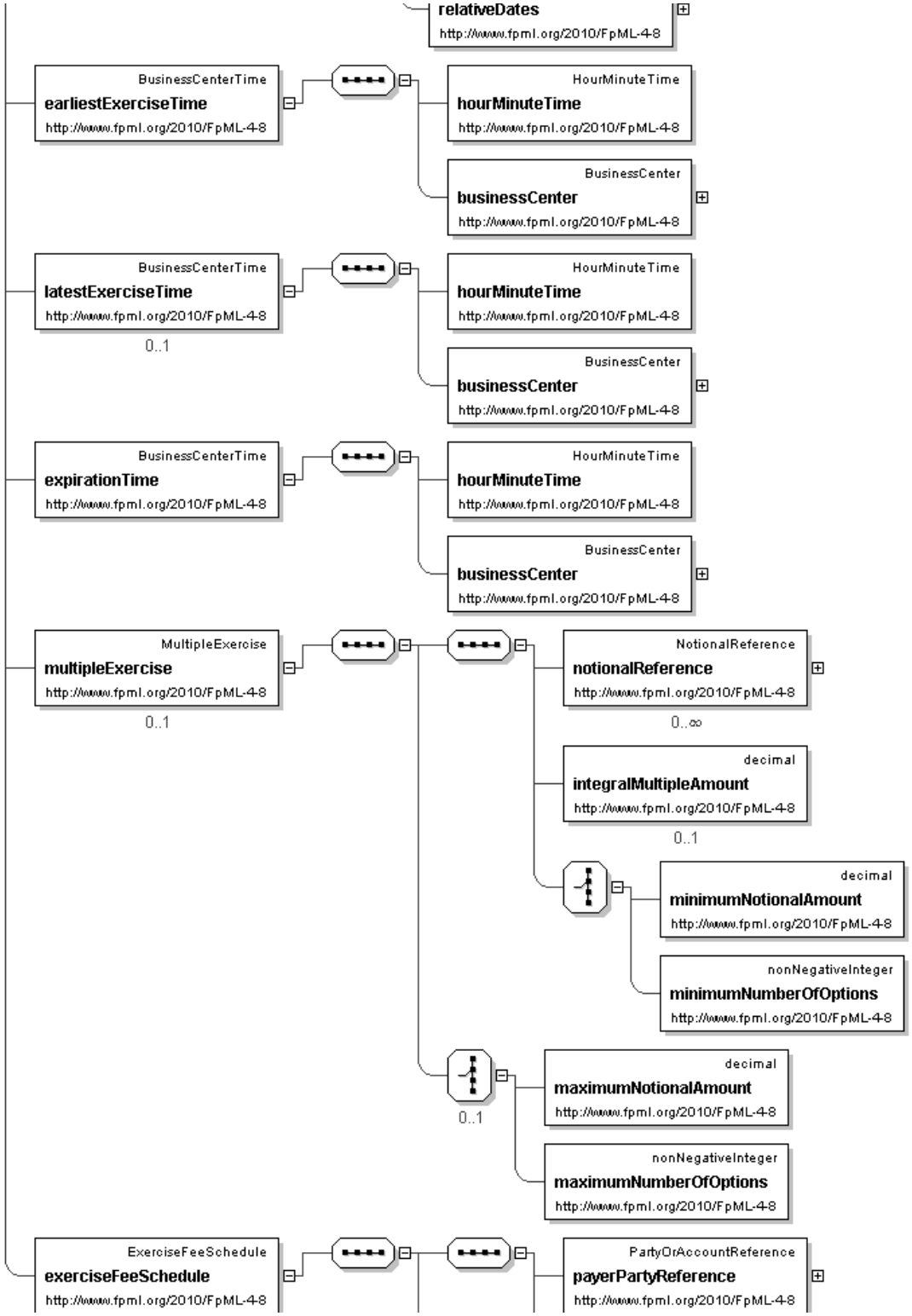
[Table of contents]

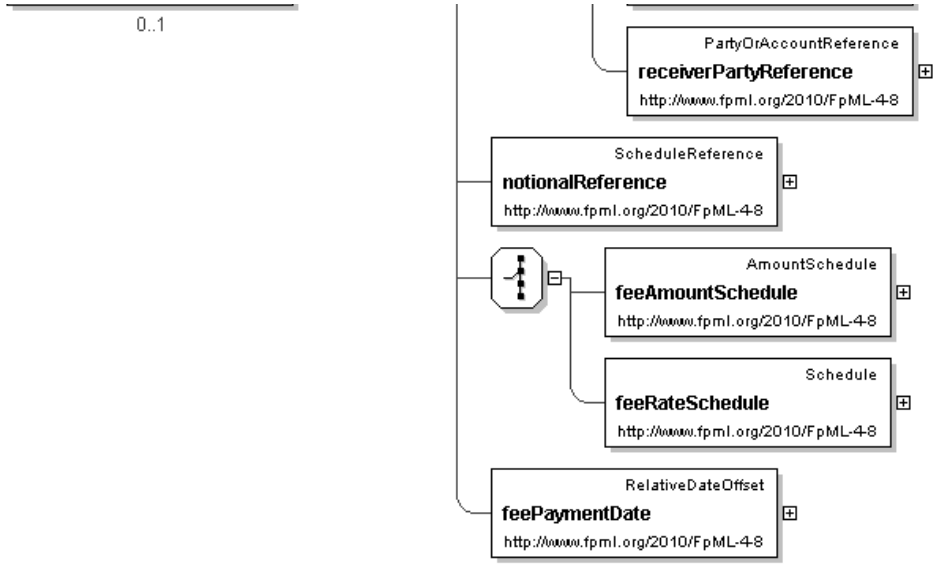
- *This element can be used wherever the following element is referenced:*
  - exercise

Name	americanExercise
Type	AmericanExercise
Nillable	no
Abstract	no
Documentation	The parameters for defining the exercise period for an American style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Logical Diagram



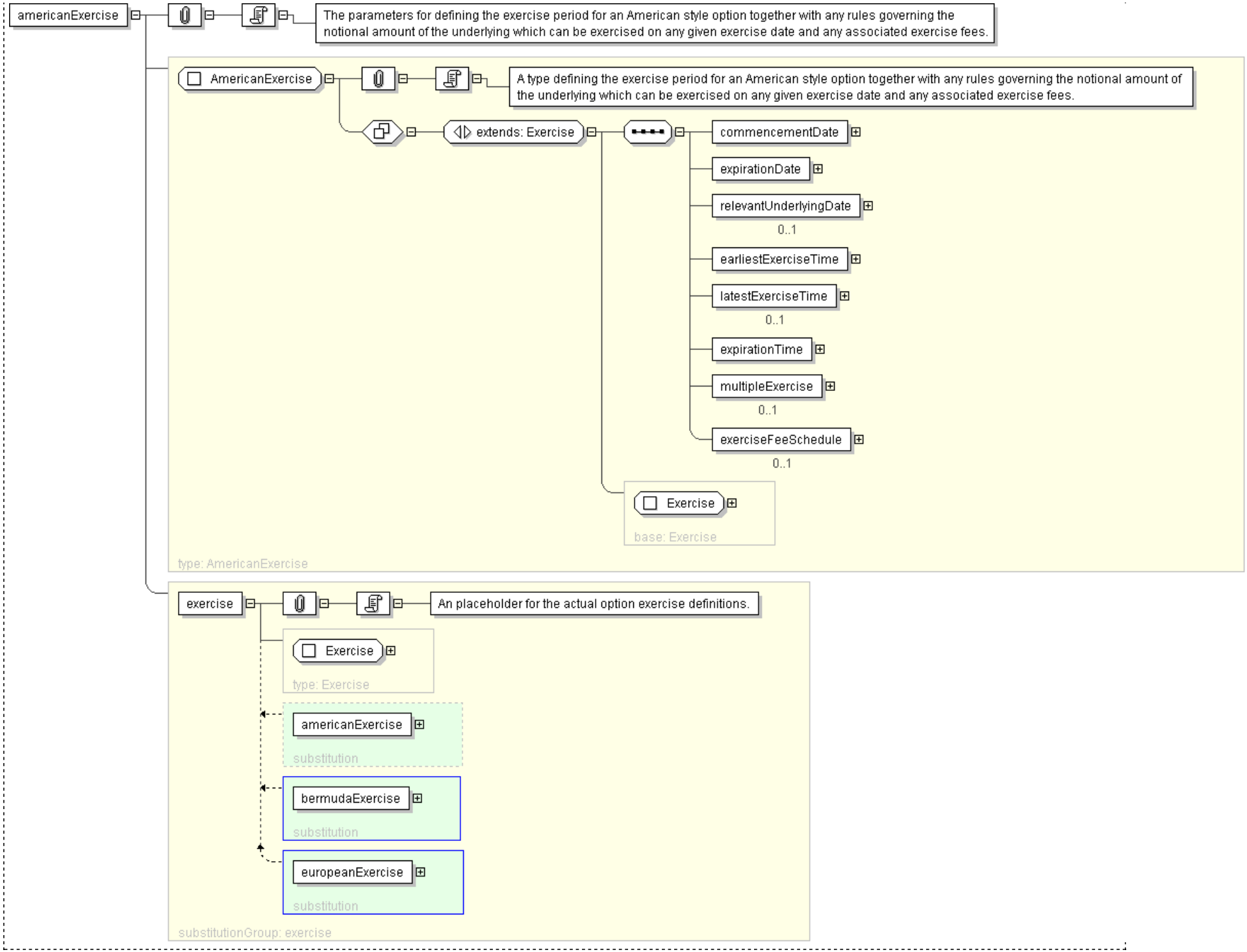

relativeDates
http://www.fpml.org/2010/FpML-4-8
BusinessCenterTime
earliestExerciseTime
http://www.fpml.org/2010/FpML-4-8
HourMinuteTime
hourMinuteTime
http://www.fpml.org/2010/FpML-4-8
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
BusinessCenterTime
latestExerciseTime
http://www.fpml.org/2010/FpML-4-8
0..1
HourMinuteTime
hourMinuteTime
http://www.fpml.org/2010/FpML-4-8
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
BusinessCenterTime
expirationTime
http://www.fpml.org/2010/FpML-4-8
HourMinuteTime
hourMinuteTime
http://www.fpml.org/2010/FpML-4-8
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
MultipleExercise
multipleExercise
http://www.fpml.org/2010/FpML-4-8
0..1
NotionalReference
notionalReference
http://www.fpml.org/2010/FpML-4-8
0..∞
decimal
integralMultipleAmount
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
minimumNotionalAmount
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
minimumNumberOfOptions
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
maximumNotionalAmount
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
maximumNumberOfOptions
http://www.fpml.org/2010/FpML-4-8
ExerciseFeeSchedule
exerciseFeeSchedule
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8



XML Instance Representation

```
<americanExercise
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
   'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the only day within the
   exercise period.'

   <relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]
   'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g. in a swaption it is
   the effective date, in an extendible/cancelable provision it is the termination date).'

   <earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]
   'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent) i) on the expriation
   date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in the case of a
   Bermuda style option the commencement date to, and including, the expiration date , in the case of an American option.'

   <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [0..1]
   'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration date) within the
   exercise period that notice can be given by the buyer to the seller or seller\'s agent. Notice of exercise given after this time
   will be deemed to have been given on the next exercise business day.'

   <expirationTime> BusinessCenterTime </expirationTime> [1]
   'The latest time for exercise on expirationDate.'

   <multipleExercise> MultipleExercise </multipleExercise> [0..1]
   'As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise all
   or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on any
   such day may not exercise less than the minimum notional amount or more that the maximum notional amount, and if an integral
   multiple amount is specified, the notional amount exercised must be equal to, or be an intergral multiple of, the integral
   multiple amount.'

   <exerciseFeeSchedule> ExerciseFeeSchedule </exerciseFeeSchedule> [0..1]
   'The fees associated with an exercise date. The fees are conditional on the exercise occuring. The fees can be specified as actual
   currency amounts or as percentages of the notional amount being exercised.'

</americanExercise>
```

Diagram



Schema Component Representation

```
<xsd:element name="americanExercise" type=" AmericanExercise " substitutionGroup="exercise"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

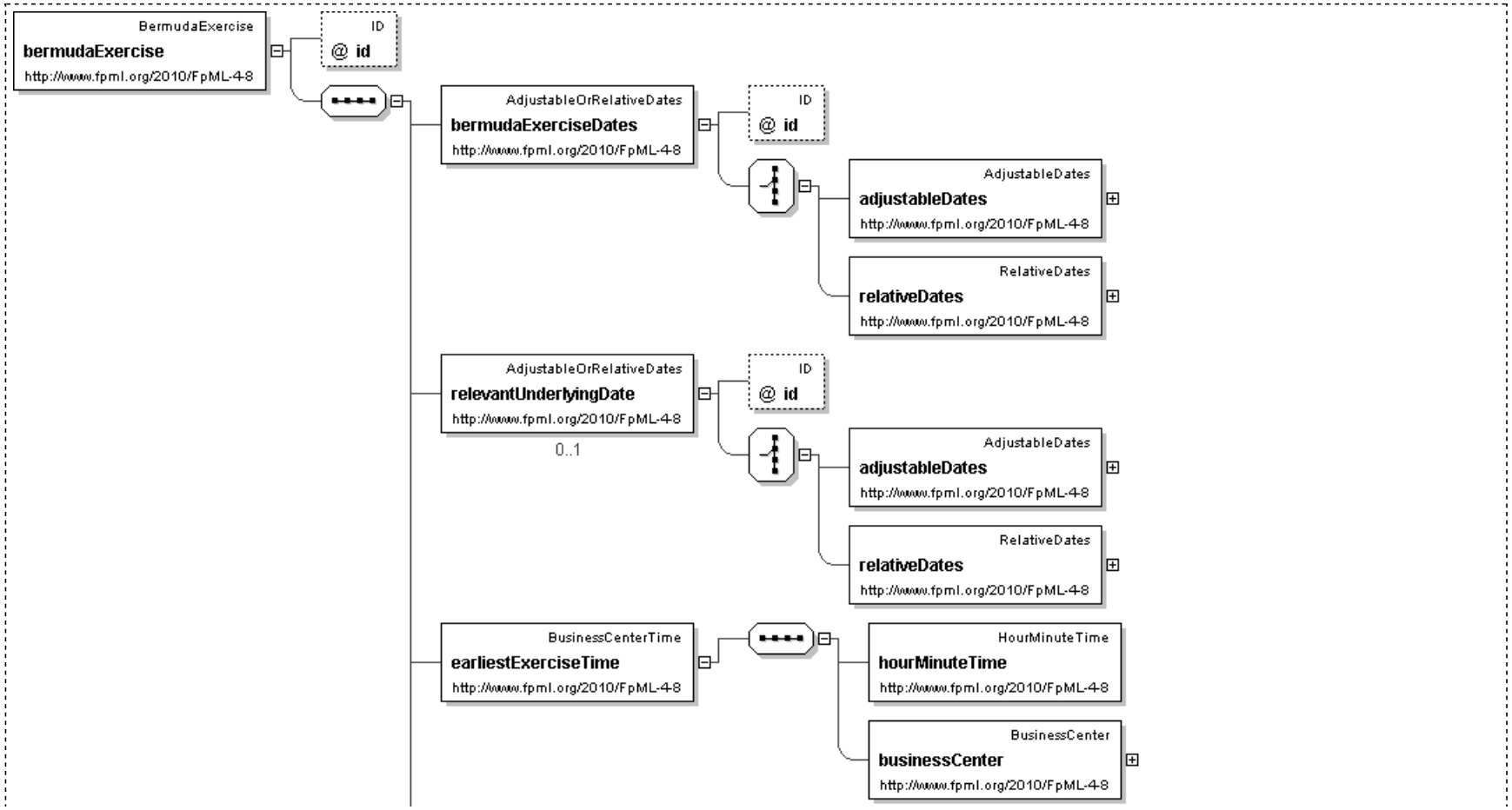
Element: bermudaExercise

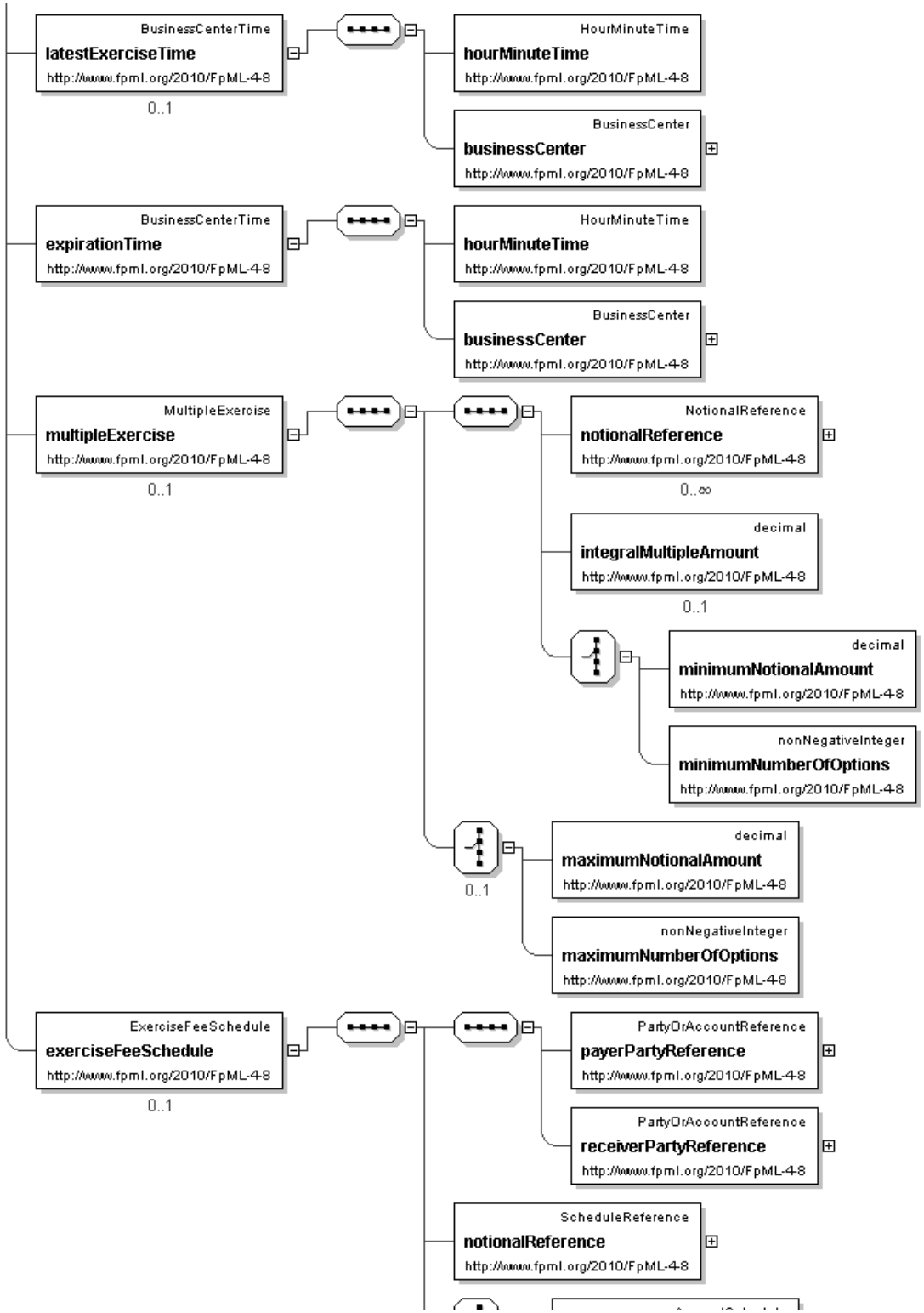
[Table of contents]

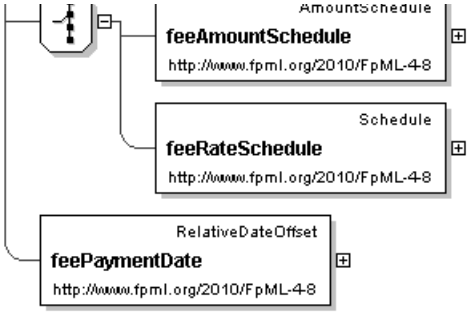
- *This element can be used wherever the following element is referenced:*
  - exercise

Name	bermudaExercise
Type	BermudaExercise
Nillable	no
Abstract	no
Documentation	The parameters for defining the exercise period for a Bermuda style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Logical Diagram



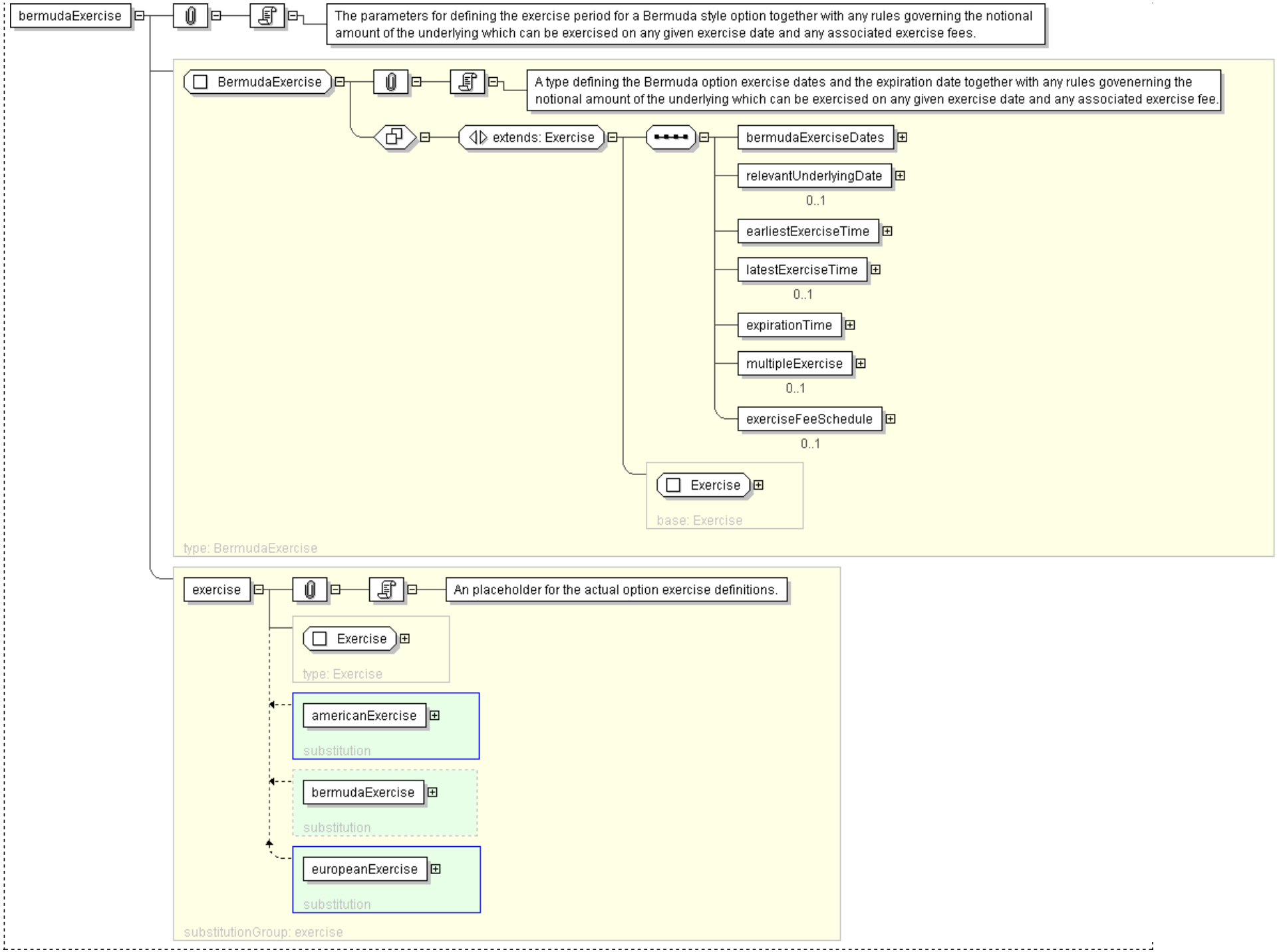

BusinessCenterTime
latestExerciseTime
http://www.fpml.org/2010/FpML-4-8
0..1
HourMinuteTime
hourMinuteTime
http://www.fpml.org/2010/FpML-4-8
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
BusinessCenterTime
expirationTime
http://www.fpml.org/2010/FpML-4-8
HourMinuteTime
hourMinuteTime
http://www.fpml.org/2010/FpML-4-8
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
MultipleExercise
multipleExercise
http://www.fpml.org/2010/FpML-4-8
0..1
NotionalReference
notionalReference
http://www.fpml.org/2010/FpML-4-8
0..∞
decimal
integralMultipleAmount
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
minimumNotionalAmount
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
minimumNumberOfOptions
http://www.fpml.org/2010/FpML-4-8
0..1
decimal
maximumNotionalAmount
http://www.fpml.org/2010/FpML-4-8
nonNegativeInteger
maximumNumberOfOptions
http://www.fpml.org/2010/FpML-4-8
ExerciseFeeSchedule
exerciseFeeSchedule
http://www.fpml.org/2010/FpML-4-8
0..1
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
ScheduleReference
notionalReference
http://www.fpml.org/2010/FpML-4-8



XML Instance Representation

```
<bermudaExercise
 id=" xsd:ID [0..1]">
   <bermudaExerciseDates> AdjustableOrRelativeDates </bermudaExerciseDates> [1]
   'The dates the define the Bermuda option exercise dates and the expiration date. The last specified date is assumed to be the
   expiration date. The dates can either be specified as a series of explicit dates and associated adjustments or as a series of
   dates defined relative to another schedule of dates, for example, the calculation period start dates. Where a relative series of
   dates are defined the first and last possible exercise dates can be separately specified.'

   <relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]
   'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g. in a swaption it is
   the effective date, in an extendible/cancelable provision it is the termination date).'

   <earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]
   'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent) i) on the
   expriation date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in
   the case of a Bermuda style option the commencement date to, and including, the expiration date , in the case of an American
   option.'

   <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [0..1]
   'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration date) within the
   exercise period that notice can be given by the buyer to the seller or seller\'s agent. Notice of exercise given after this
   time will be deemed to have been given on the next exercise business day.'

   <expirationTime> BusinessCenterTime </expirationTime> [1]
   'The latest time for exercise on expirationDate.'

   <multipleExercise> MultipleExercise </multipleExercise> [0..1]
   'As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise
   all or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on
   any such day may not exercise less than the minimum notional amount or more that the maximum notional amount, and if an
   integral multiple amount is specified, the notional amount exercised must be equal to, or be an intergral multiple of, the
   integral multiple amount.'

   <exerciseFeeSchedule> ExerciseFeeSchedule </exerciseFeeSchedule> [0..1]
   'The fees associated with an exercise date. The fees are conditional on the exercise occuring. The fees can be specified as
   actual currency amounts or as percentages of the notional amount being exercised.'

</bermudaExercise>
```

Diagram



Schema Component Representation

```
<xsd:element name="bermudaExercise" type=" BermudaExercise " substitutionGroup="exercise"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

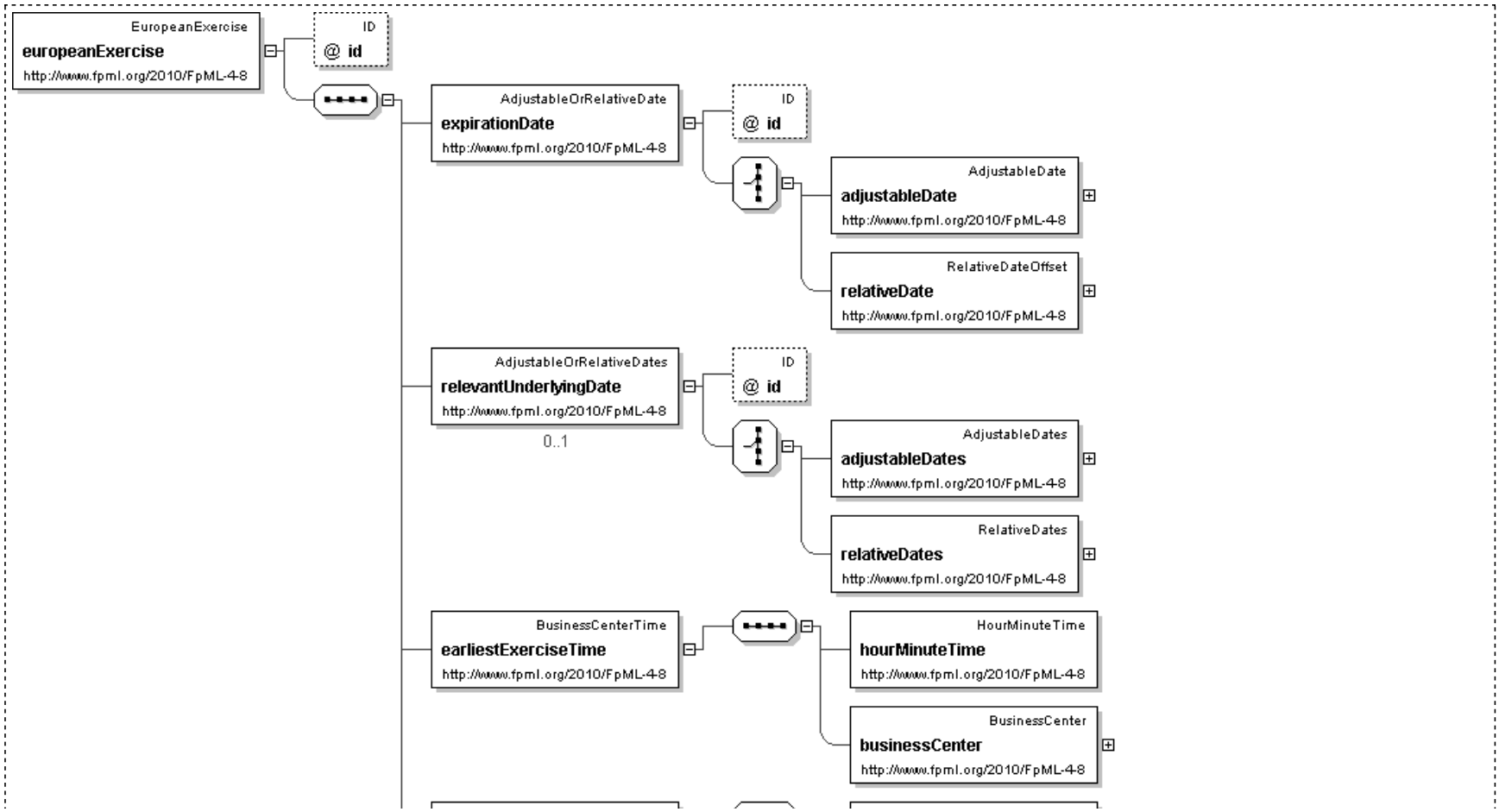
Element: europeanExercise

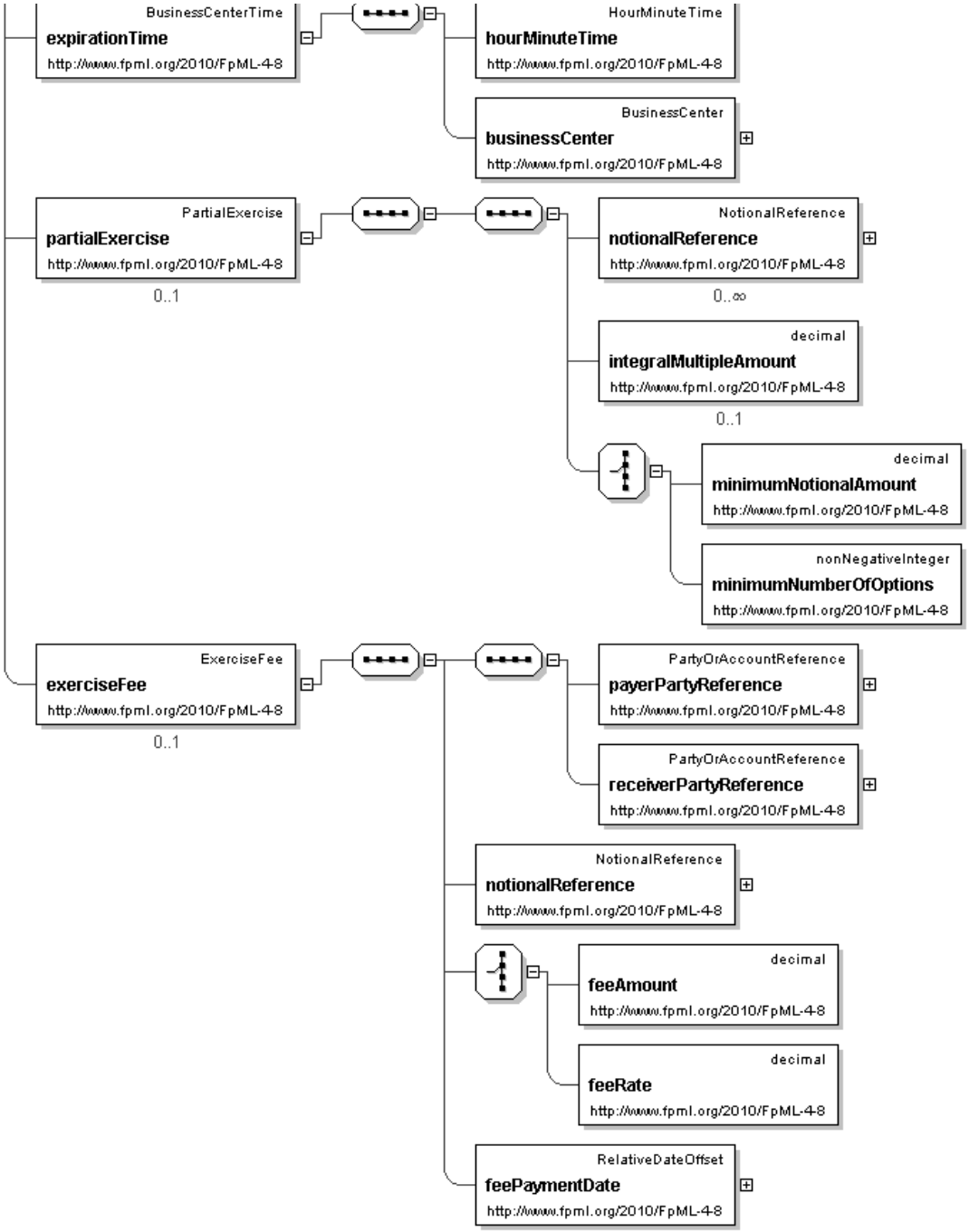
[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - exercise

Name	europeanExercise
Type	EuropeanExercise
Nillable	no
Abstract	no
Documentation	The parameters for defining the exercise period for a European style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Logical Diagram





XML Instance Representation

```
<europeanExercise
 id=" xsd:ID [0..1]">
   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the only day within the
   exercise period.'
```

<relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]

'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g. in a swaption it is the effective date, in an extendible/cancelable provision it is the termination date).'

<earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]

'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent) i) on the expriation date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in the case of a Bermuda style option the commencement date to, and including, the expiration date , in the case of an American option.'

<expirationTime> BusinessCenterTime </expirationTime> [1]

'The latest time for exercise on expirationDate.'

<partialExercise> PartialExercise </partialExercise> [0..1]

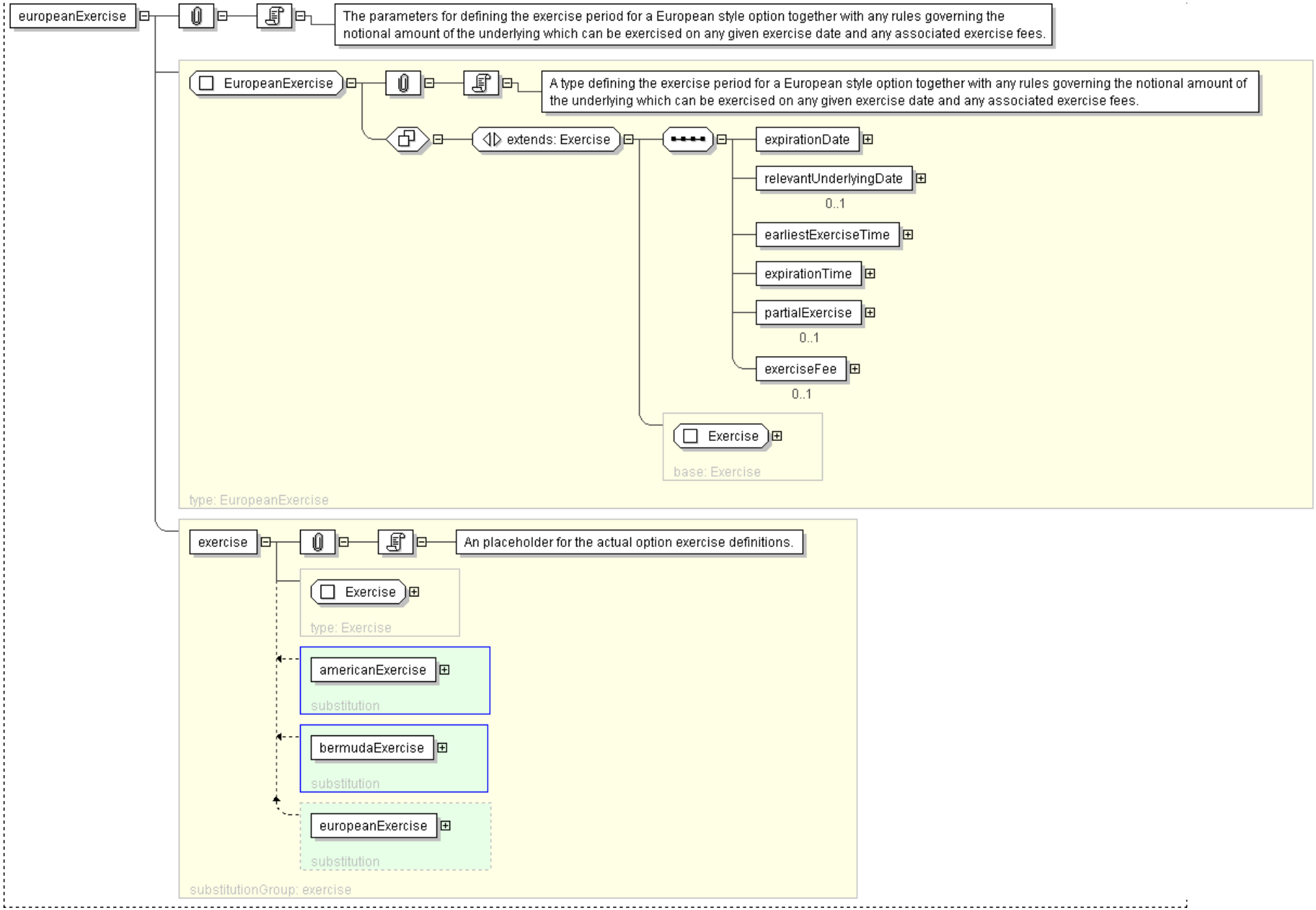
'As defined in the 2000 ISDA Definitions, Section 12.3. Partial Exercise, the buyer of the option has the right to exercise all or less than all the notional amount of the underlying swap on the expiration date, but may not exercise less than the minimum notional amount, and if an integral multiple amount is specified, the notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.'

<exerciseFee> ExerciseFee </exerciseFee> [0..1]

'A fee to be paid on exercise. This could be represented as an amount or a rate and notional reference on which to apply the rate.'

</europeanExercise>

Diagram



Schema Component Representation

```
<xsd:element name="europeanExercise" type=" EuropeanExercise " substitutionGroup="exercise"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: exercise

[Table of contents]

- *The following elements can be used wherever this element is referenced:*
  - americanExercise
  - bermudaExercise
  - europeanExercise

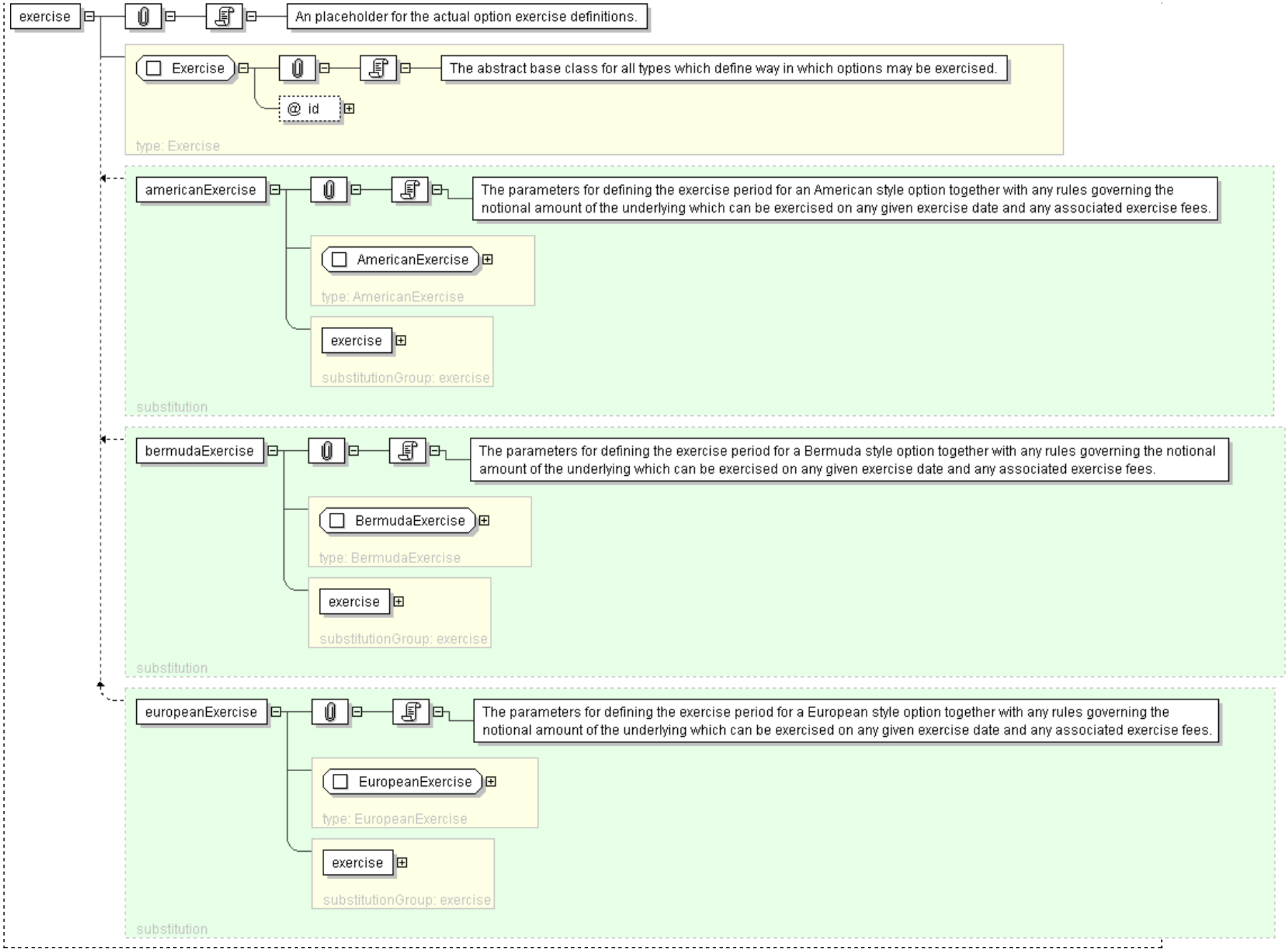
Name	exercise
Type	Exercise
Nillable	no
Abstract	yes
Documentation	An placeholder for the actual option exercise definitions.

Logical Diagram

XML Instance Representation

```
<exercise
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:element name="exercise" type=" Exercise " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

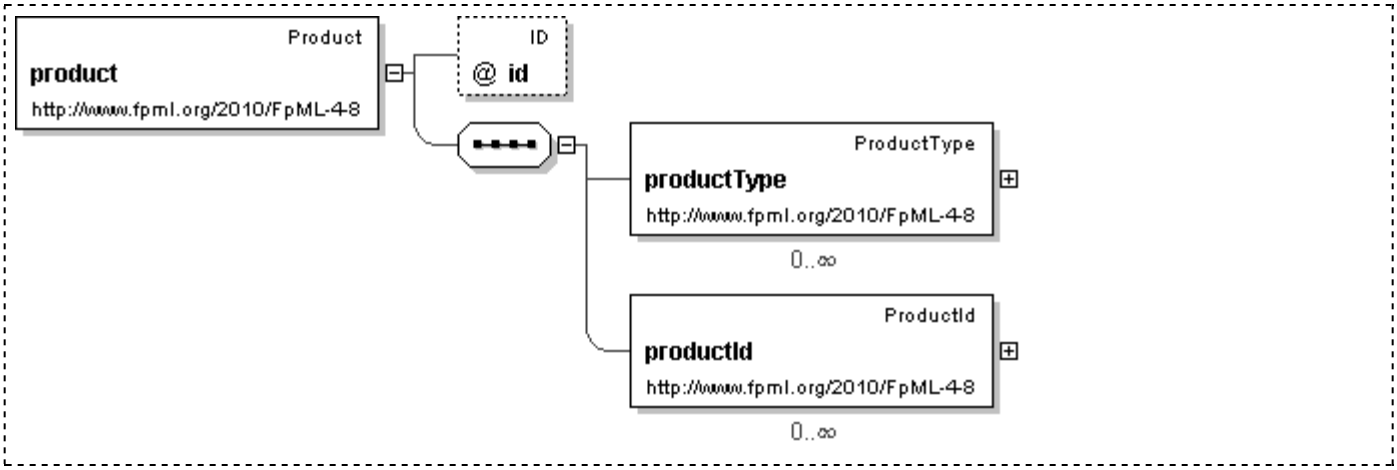
XML Schema Documentation

Element: product

[Table of contents]

Name	product
Type	Product
Nillable	no
Abstract	yes
Documentation	An abstract element used as a place holder for the substituting product elements.

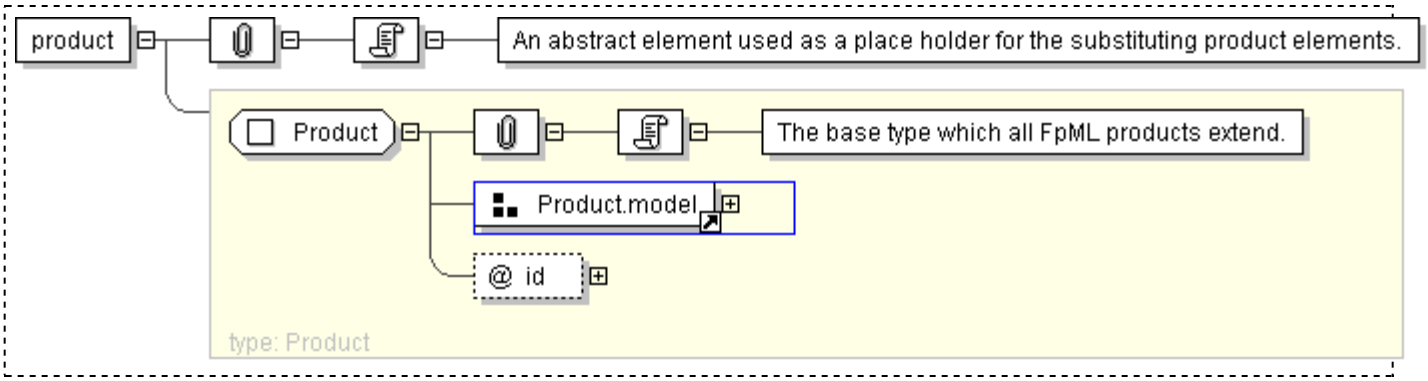
Logical Diagram



XML Instance Representation

```
<product
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

</product>
```

Diagram



Schema Component Representation

```
<xsd:element name="product" type=" Product " abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BusinessCentersOrReference.model

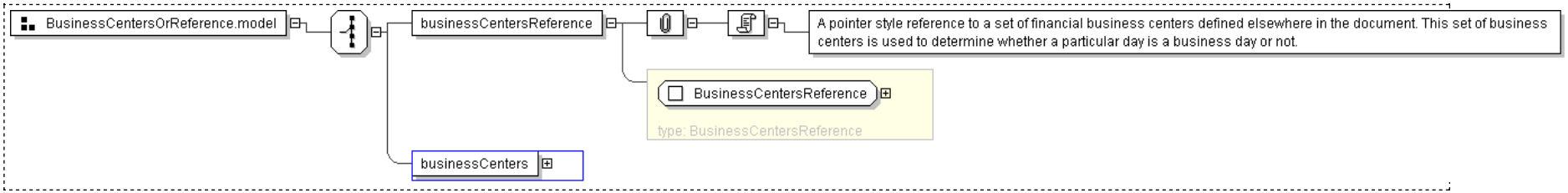
[Table of contents]

Name	BusinessCentersOrReference.model
Used by (from the same schema document)	Complex Type BusinessDateRange , Complex Type BusinessDayAdjustments , Complex Type RelativeDateOffset , Complex Type RelativeDateSequence

XML Instance Representation

```
Start Choice [1]
   <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
   'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to determine
   whether a particular day is a business day or not.'

   <businessCenters> BusinessCenters </businessCenters> [1]
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="BusinessCentersOrReference.model">
   <xsd:choice>
      <xsd:element name="businessCentersReference" type=" BusinessCentersReference "/>
      <xsd:element name="businessCenters" type=" BusinessCenters "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: BuyerSeller.model

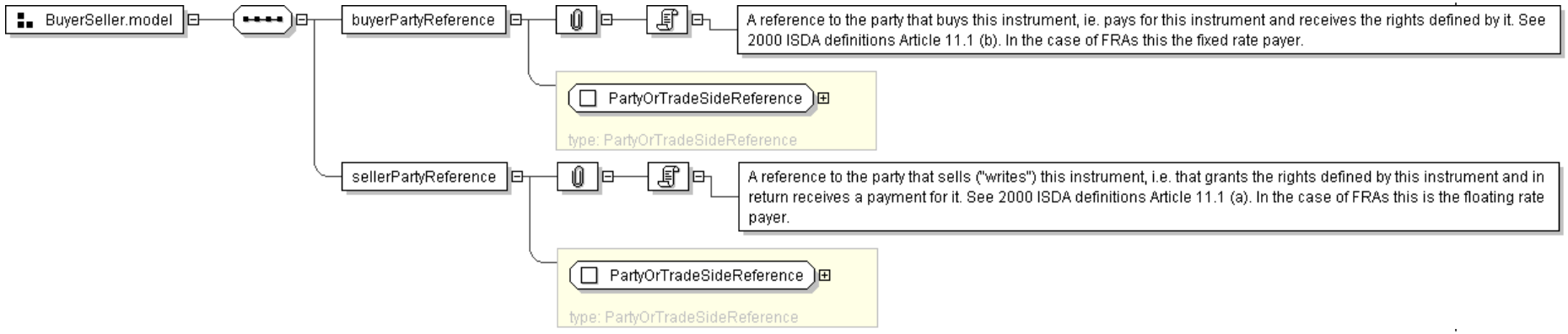
[Table of contents]

Name	BuyerSeller.model

XML Instance Representation

```
<buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA
definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

<sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return
receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'
```

Diagram



Schema Component Representation

```
<xsd:group name="BuyerSeller.model">
  <xsd:sequence>
    <xsd:element name="buyerPartyReference" type=" PartyOrTradeSideReference "/>
    <xsd:element name="sellerPartyReference" type=" PartyOrTradeSideReference "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: FloatingRateIndex.model

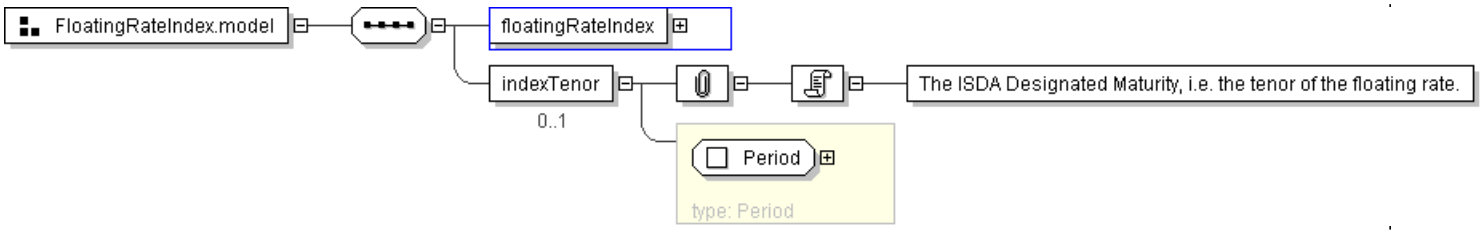
[Table of contents]

Name	FloatingRateIndex.model
Used by (from the same schema document)	Complex Type FloatingRate

XML Instance Representation

```
<floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
<indexTenor> Period </indexTenor> [0..1]
'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'
```

Diagram



Schema Component Representation

```
<xsd:group name="FloatingRateIndex.model">
  <xsd:sequence>
    <xsd:element name="floatingRateIndex" type=" FloatingRateIndex "/>
    <xsd:element name="indexTenor" type=" Period " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PartialExercise.model

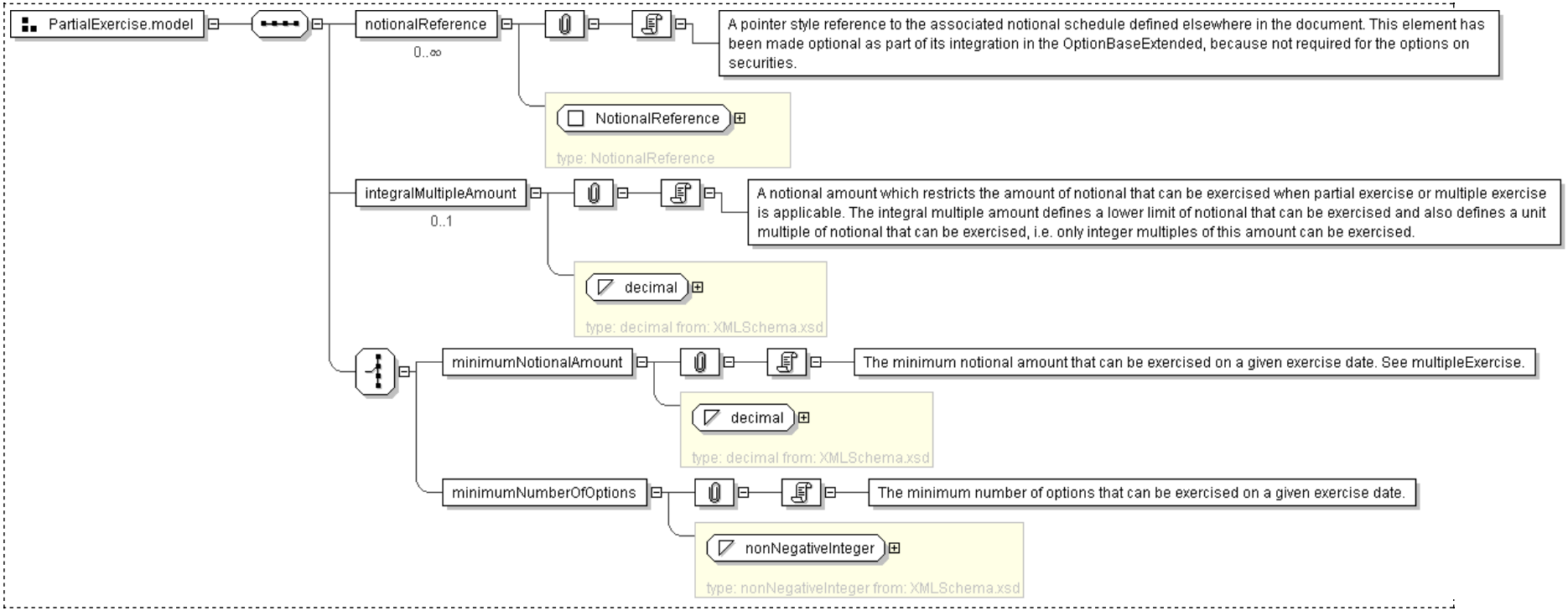
[Table of contents]

Name	PartialExercise.model
Used by (from the same schema document)	Complex Type MultipleExercise , Complex Type PartialExercise

XML Instance Representation

```
<notionalReference> NotionalReference </notionalReference> [0..*]
'A pointer style reference to the associated notional schedule defined elsewhere in the document. This element has been made optional as part of
its integration in the OptionBaseExtended, because not required for the options on securities.'

<integralMultipleAmount> xsd:decimal </integralMultipleAmount> [0..1]
'A notional amount which restricts the amount of notional that can be exercised when partial exercise or multiple exercise is applicable. The
integral multiple amount defines a lower limit of notional that can be exercised and also defines a unit multiple of notional that can be
exercised, i.e. only integer multiples of this amount can be exercised.'

Start Choice [1]
   <minimumNotionalAmount> xsd:decimal </minimumNotionalAmount> [1]
   'The minimum notional amount that can be exercised on a given exercise date. See multipleExercise.'

   <minimumNumberOfOptions> xsd:nonNegativeInteger </minimumNumberOfOptions> [1]
   'The minimum number of options that can be exercised on a given exercise date.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="PartialExercise.model">
  <xsd:sequence>
    <xsd:element name="                  " type="                  " minOccurs="  " maxOccurs="         "/>
```

```
                        notionalReference         NotionalReference              0              unbounded
     <xsd:element name="integralMultipleAmount" type=" xsd:decimal " minOccurs="0"/>
     <xsd:choice>
        <xsd:element name="minimumNotionalAmount" type=" xsd:decimal "/>
        <xsd:element name="minimumNumberOfOptions" type=" xsd:nonNegativeInteger "/>
     </xsd:choice>
   </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PayerReceiver.model

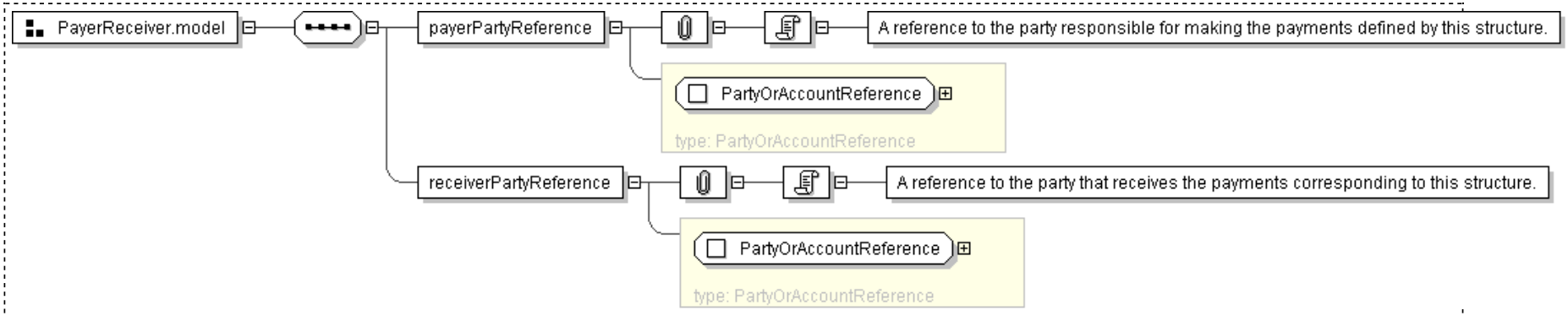
[Table of contents]

Name	PayerReceiver.model
Used by (from the same schema document)	Complex Type ExerciseFee , Complex Type ExerciseFeeSchedule , Complex Type Payment , Complex Type PaymentBaseExtended , Complex Type SimplePayment

XML Instance Representation

```
<payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
'A reference to the party responsible for making the payments defined by this structure.'

<receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
'A reference to the party that receives the payments corresponding to this structure.'
```

Diagram



Schema Component Representation

```
<xsd:group name="PayerReceiver.model">
  <xsd:sequence>
    <xsd:element name="payerPartyReference" type=" PartyOrAccountReference "/>
    <xsd:element name="receiverPartyReference" type=" PartyOrAccountReference "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: PaymentDiscounting.model

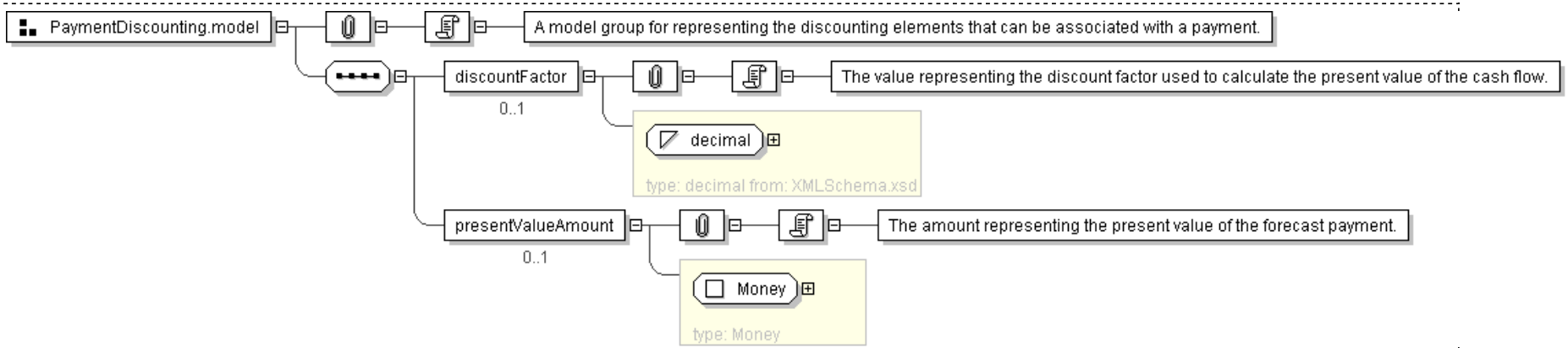
[Table of contents]

Name	PaymentDiscounting.model
Documentation	A model group for representing the discounting elements that can be associated with a payment.

XML Instance Representation

```
<discountFactor> xsd:decimal </discountFactor> [0..1]
'The value representing the discount factor used to calculate the present value of the cash flow.'

<presentValueAmount> Money </presentValueAmount> [0..1]
'The amount representing the present value of the forecast payment.'
```

Diagram



Schema Component Representation

```
<xsd:group name="PaymentDiscounting.model">
  <xsd:sequence>
    <xsd:element name="discountFactor" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="presentValueAmount" type=" Money " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Period.model

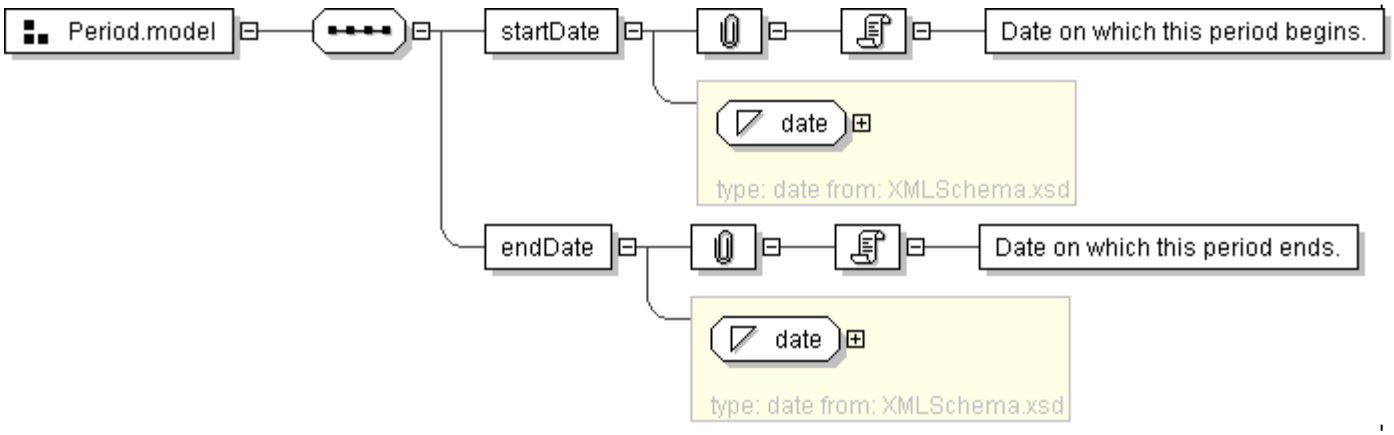
[Table of contents]

Name	Period.model

XML Instance Representation

```
<startDate> xsd:date </startDate> [1]
'Date on which this period begins.'

<endDate> xsd:date </endDate> [1]
'Date on which this period ends.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Period.model">
  <xsd:sequence>
    <xsd:element name="startDate" type=" xsd:date "/>
    <xsd:element name="endDate" type=" xsd:date "/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Premium.model

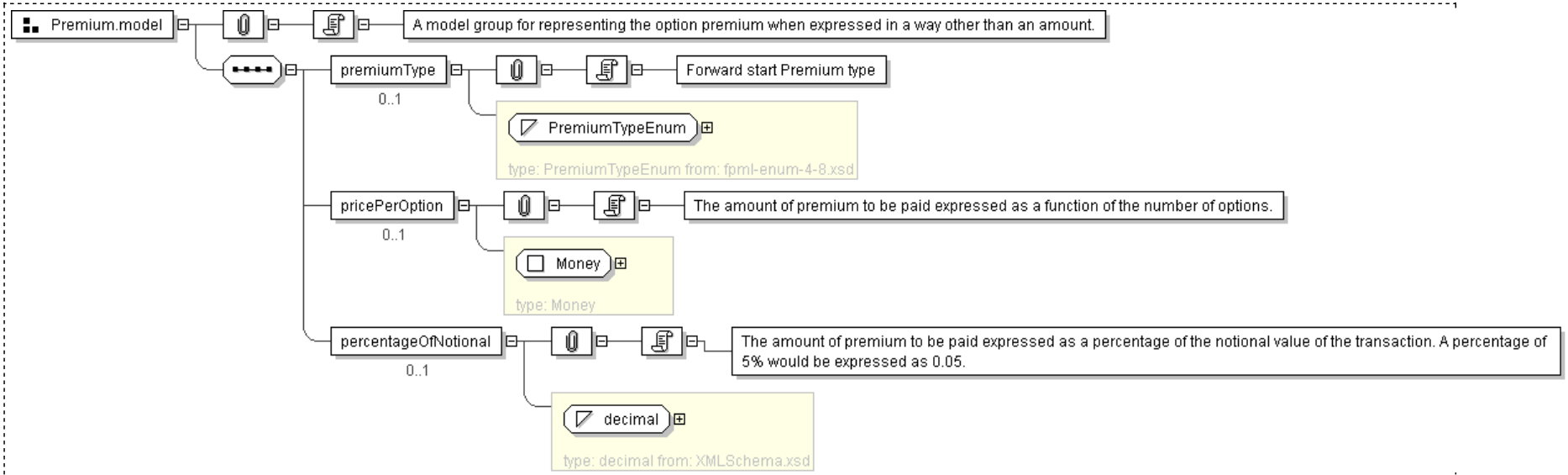
[Table of contents]

Name	Premium.model
Documentation	A model group for representing the option premium when expressed in a way other than an amount.

XML Instance Representation

```
<premiumType> PremiumTypeEnum </premiumType> [0..1]
'Forward start Premium type'

<pricePerOption> Money </pricePerOption> [0..1]
'The amount of premium to be paid expressed as a function of the number of options.'

<percentageOfNotional> xsd:decimal </percentageOfNotional> [0..1]
'The amount of premium to be paid expressed as a percentage of the notional value of the transaction. A percentage of 5% would be expressed
as 0.05.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Premium.model">
  <xsd:sequence>
    <xsd:element name="premiumType" type=" PremiumTypeEnum " minOccurs="0"/>
    <xsd:element name="pricePerOption" type=" Money " minOccurs="0"/>
    <xsd:element name="percentageOfNotional" type=" xsd:decimal " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: Product.model

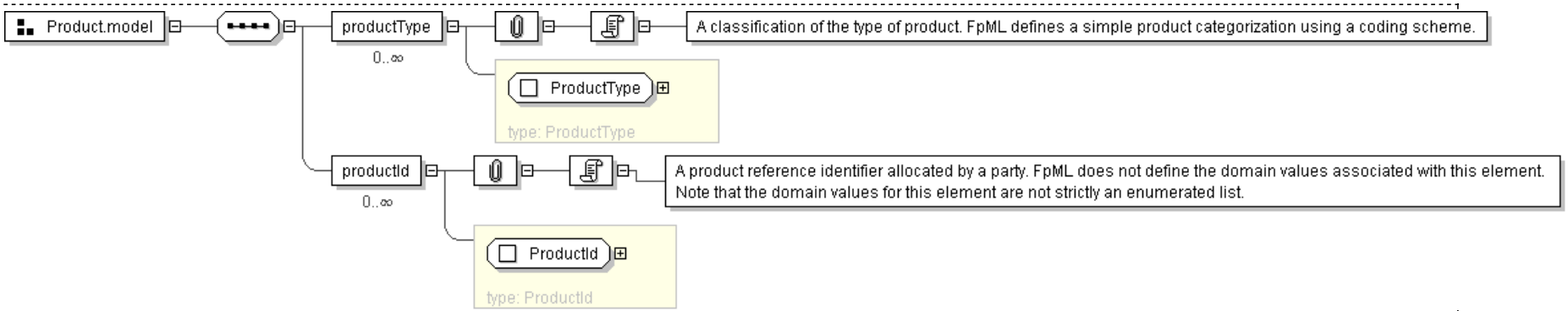
[Table of contents]

Name	Product.model
Used by (from the same schema document)	Complex Type Product

XML Instance Representation

```
<productType> ProductType </productType> [0..*]
'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

<productId> ProductId </productId> [0..*]
'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that
the domain values for this element are not strictly an enumerated list.'
```

Diagram



Schema Component Representation

```
<xsd:group name="Product.model">
  <xsd:sequence>
    <xsd:element name="productType" type=" ProductType " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="productId" type=" ProductId " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: RoutingExplicitDetails.model

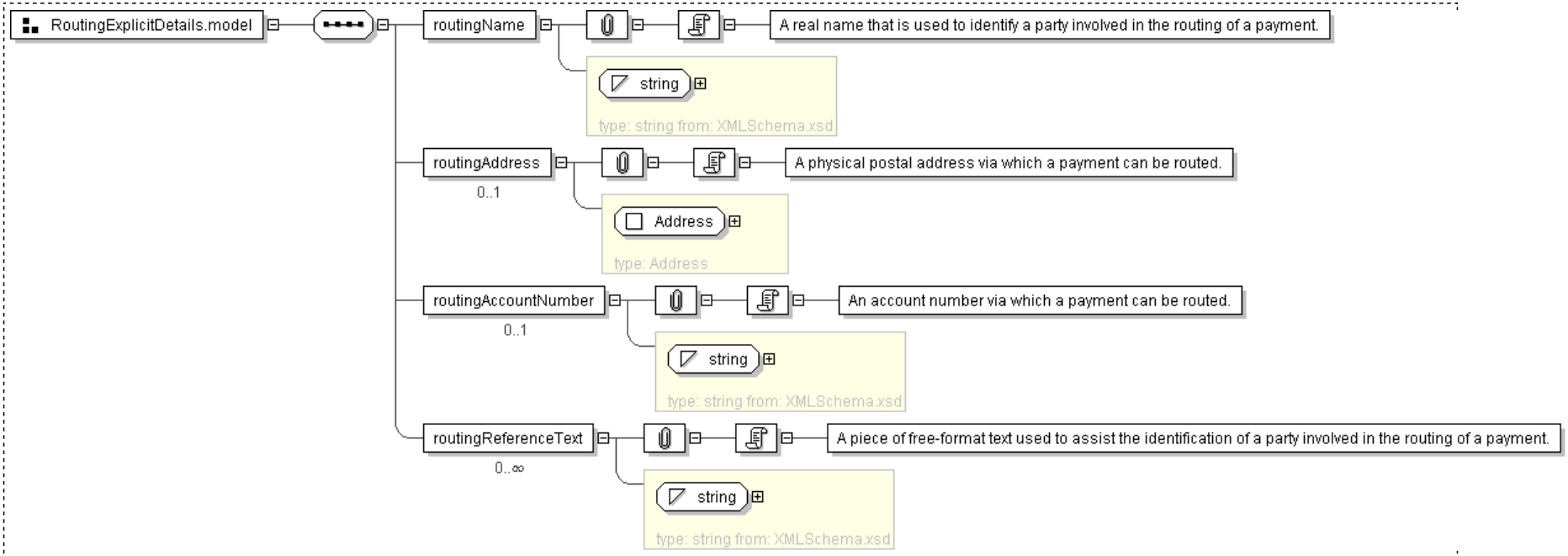
[Table of contents]

Name	RoutingExplicitDetails.model
Used by (from the same schema document)	Complex Type RoutingExplicitDetails , Complex Type RoutingIdsAndExplicitDetails

XML Instance Representation

```
<routingName> xsd:string </routingName> [1]
'A real name that is used to identify a party involved in the routing of a payment.'

<routingAddress> Address </routingAddress> [0..1]
'A physical postal address via which a payment can be routed.'

<routingAccountNumber> xsd:string </routingAccountNumber> [0..1]
'An account number via which a payment can be routed.'

<routingReferenceText> xsd:string </routingReferenceText> [0..*]
'A piece of free-format text used to assist the identification of a party involved in the routing of a payment.'
```

Diagram



Schema Component Representation

```
<xsd:group name="RoutingExplicitDetails.model">
  <xsd:sequence>
    <xsd:element name="routingName" type=" xsd:string "/>
    <xsd:element name="routingAddress" type=" Address " minOccurs="0"/>
    <xsd:element name="routingAccountNumber" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="routingReferenceText" type=" xsd:string " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: RoutingIdentification.model

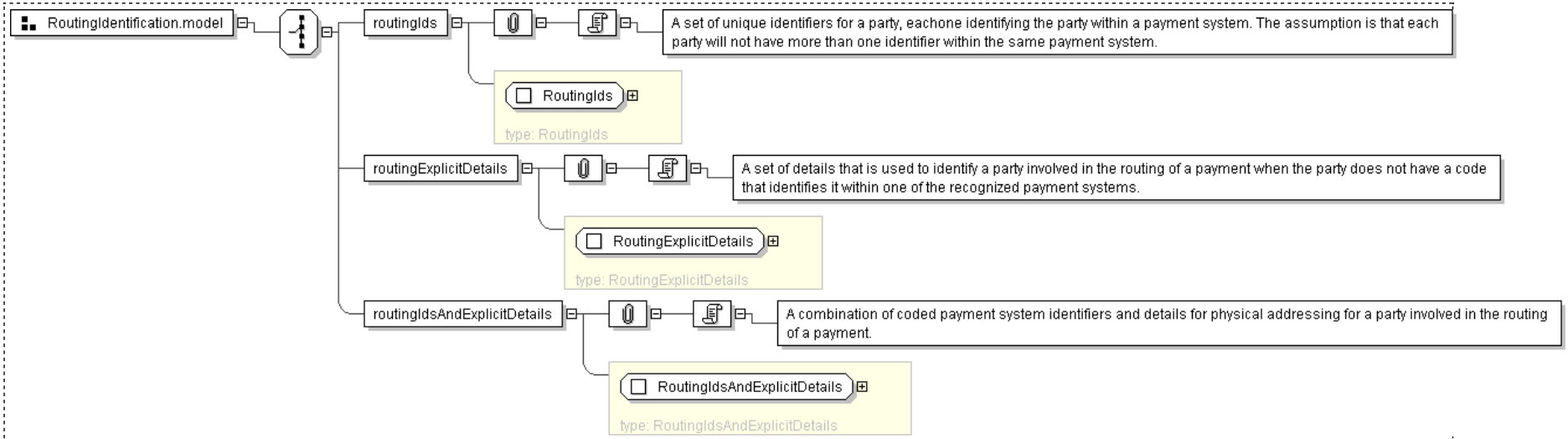
[Table of contents]

Name	RoutingIdentification.model
Used by (from the same schema document)	Complex Type Beneficiary , Complex Type CorrespondentInformation , Complex Type IntermediaryInformation , Complex Type Routing

XML Instance Representation

```
Start Choice [1]
   <routingIds> RoutingIds </routingIds> [1]
    'A set of unique identifiers for a party, eachone identifying the party within a payment system. The assumption is that each party will not have
    more than one identifier within the same payment system.'

   <routingExplicitDetails> RoutingExplicitDetails </routingExplicitDetails> [1]
    'A set of details that is used to identify a party involved in the routing of a payment when the party does not have a code that identifies it
    within one of the recognized payment systems.'

   <routingIdsAndExplicitDetails> RoutingIdsAndExplicitDetails </routingIdsAndExplicitDetails> [1]
    'A combination of coded payment system identifiers and details for physical addressing for a party involved in the routing of a payment.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="RoutingIdentification.model">
  <xsd:choice>
    <xsd:element name="routingIds" type=" RoutingIds "/>
    <xsd:element name="routingExplicitDetails" type=" RoutingExplicitDetails "/>
    <xsd:element name="routingIdsAndExplicitDetails" type=" RoutingIdsAndExplicitDetails "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: SettlementAmountOrCurrency.model

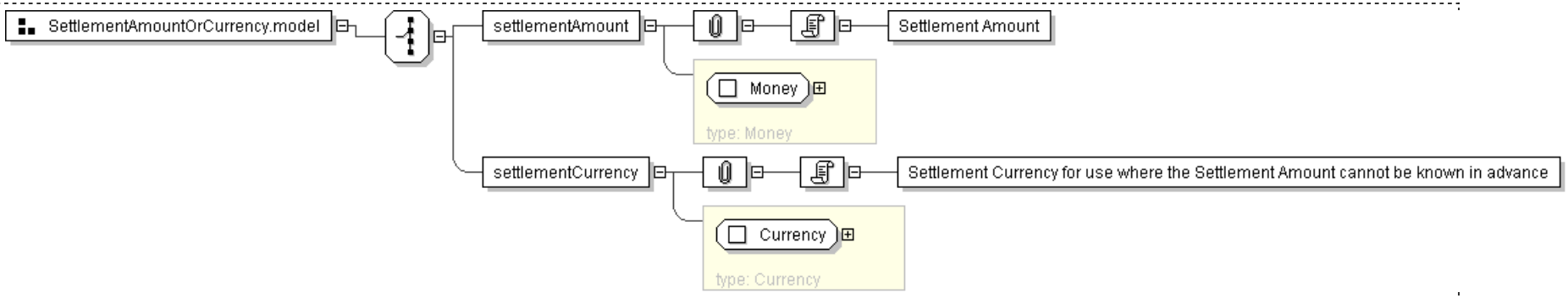
[Table of contents]

Name	SettlementAmountOrCurrency.model

XML Instance Representation

```
Start Choice [1]
   <settlementAmount> Money </settlementAmount> [1]
   'Settlement Amount'

   <settlementCurrency> Currency </settlementCurrency> [1]
   'Settlement Currency for use where the Settlement Amount cannot be known in advance'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="SettlementAmountOrCurrency.model">
  <xsd:choice>
    <xsd:element name="settlementAmount" type=" Money "/>
    <xsd:element name="settlementCurrency" type=" Currency "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: VersionHistory.model

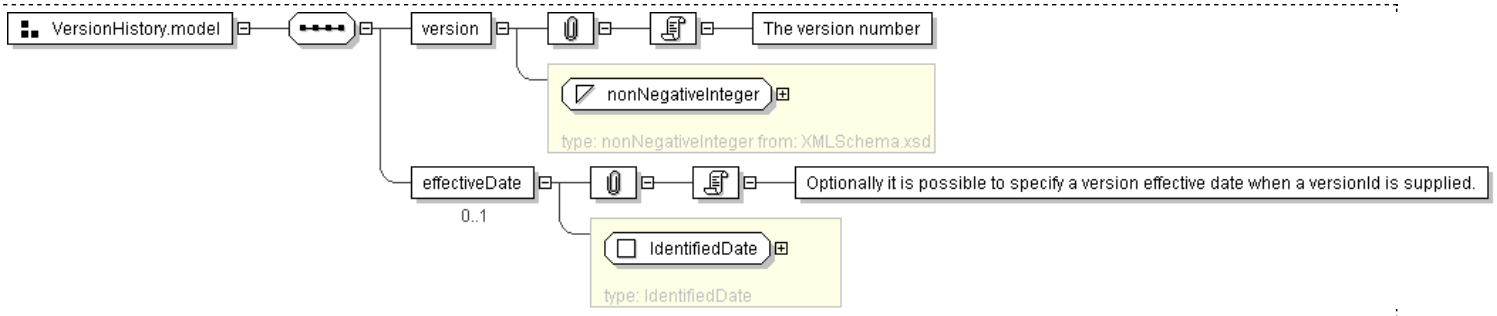
[Table of contents]

Name	VersionHistory.model

XML Instance Representation

```
<version> xsd:nonNegativeInteger </version> [1]
'The version number'

<effectiveDate> IdentifiedDate </effectiveDate> [0..1]
'Optionally it is possible to specify a version effective date when a versionId is supplied.'
```

Diagram



Schema Component Representation

```
<xsd:group name="VersionHistory.model">
  <xsd:sequence>
    <xsd:element name="version" type=" xsd:nonNegativeInteger "/>
    <xsd:element name="effectiveDate" type=" IdentifiedDate " minOccurs="0"/>
  </xsd:sequence>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Account

[Table of contents]

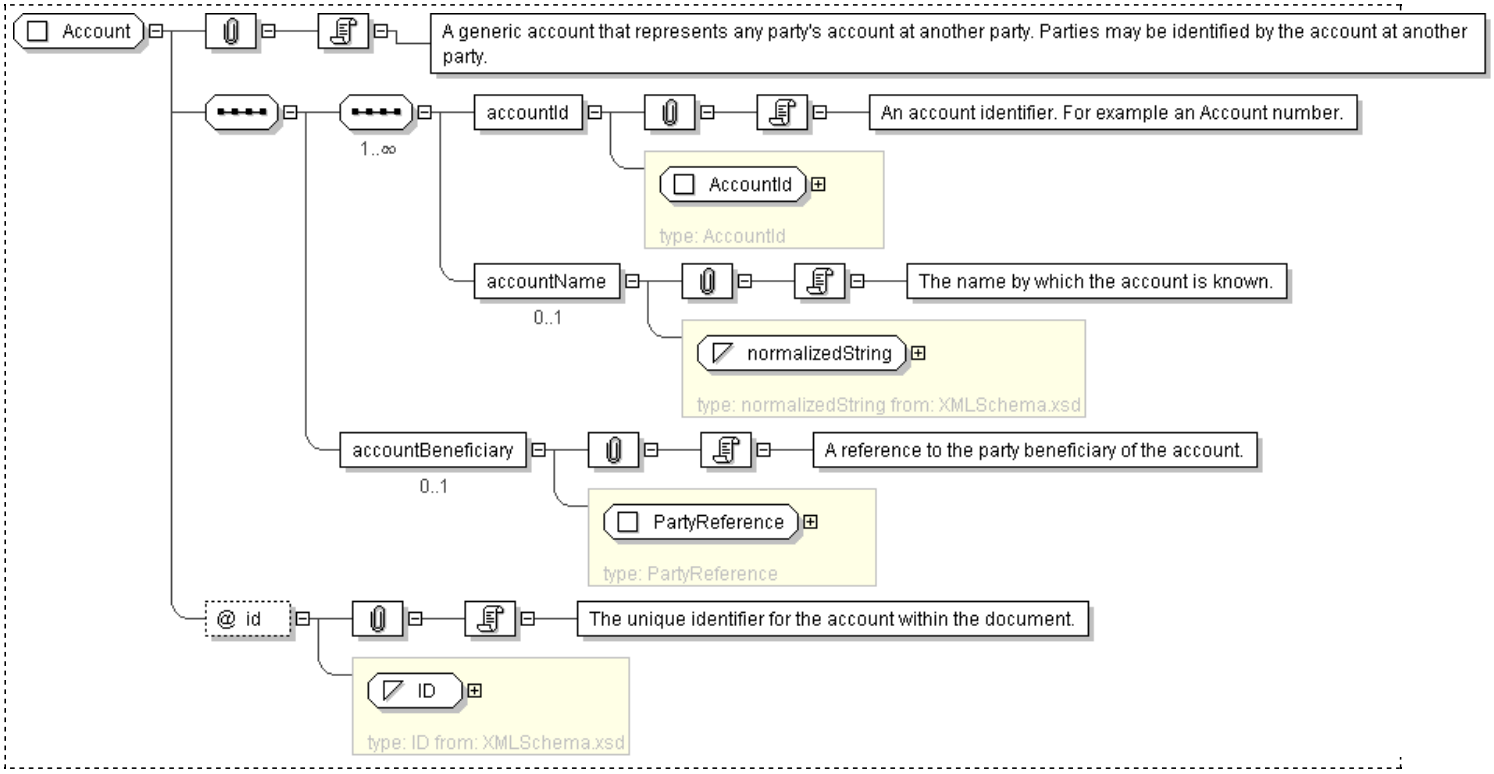
Super-types:	None
Sub-types:	None

Name	Account
Used by (from the same schema document)	Complex Type Party
Abstract	no
Documentation	A generic account that represents any party's account at another party. Parties may be identified by the account at another party.

XML Instance Representation

```
<...
 id=" xsd:ID [1]
 'The unique identifier for the account within the document.'

 ">
   Start Sequence [1..*]
      <accountId> AccountId </accountId> [1]
      'An account identifier. For example an Account number.'

      <accountName> xsd:normalizedString </accountName> [0..1]
      'The name by which the account is known.'

   End Sequence
   <accountBeneficiary> PartyReference </accountBeneficiary> [0..1]
   'A reference to the party beneficiary of the account.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Account">
  <xsd:sequence>
    <xsd:sequence maxOccurs="unbounded">
      <xsd:element name="accountId" type=" AccountId "/>
      <xsd:element name="accountName" type=" xsd:normalizedString " minOccurs="0"/>
    </xsd:sequence>
```

```
      <xsd:element name="accountBeneficiary" type=" PartyReference " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AccountId

[Table of contents]

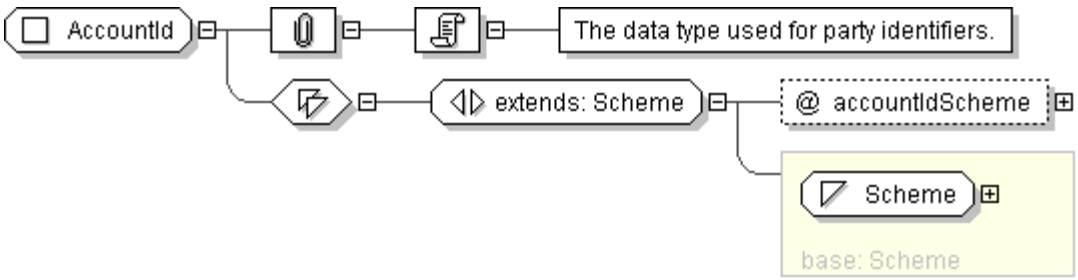
Super-types:	xsd:normalizedString < Scheme (by restriction) < AccountId (by extension)
Sub-types:	None

Name	AccountId
Used by (from the same schema document)	Complex Type Account
Abstract	no
Documentation	The data type used for party identifiers.

XML Instance Representation

```
<...
 accountIdScheme=" xsd:anyURI [0..1]
 'The identifier scheme used with this accountId. A unique URI to determine the
 authoritative issuer of these identifiers.'

 ">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AccountId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="accountIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AccountReference

[Table of contents]

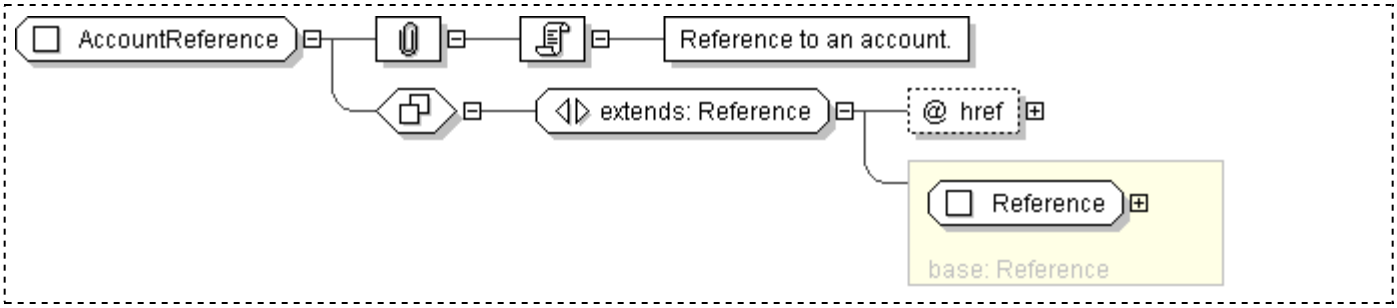
Super-types:	Reference < AccountReference (by extension)
Sub-types:	None

Name	AccountReference
Abstract	no
Documentation	Reference to an account.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AccountReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Account"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Address

[Table of contents]

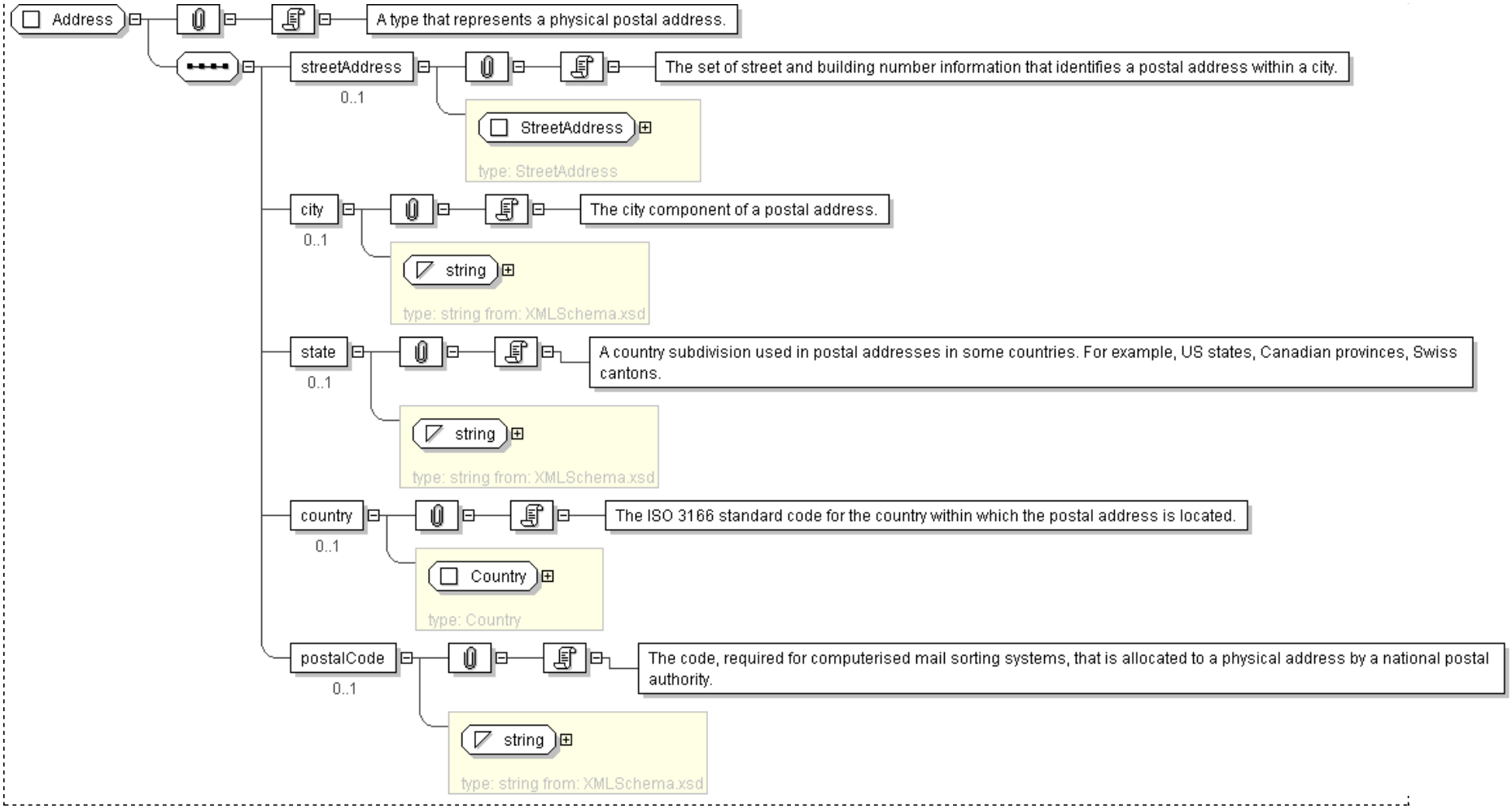
Super-types:	None
Sub-types:	None

Name	Address
Used by (from the same schema document)	Model Group RoutingExplicitDetails.model
Abstract	no
Documentation	A type that represents a physical postal address.

XML Instance Representation

```
<...>
   <streetAddress> StreetAddress </streetAddress> [0..1]
   'The set of street and building number information that identifies a postal address within a city.'

   <city> xsd:string </city> [0..1]
   'The city component of a postal address.'

   <state> xsd:string </state> [0..1]
   'A country subdivision used in postal addresses in some countries. For example, US states, Canadian provinces, Swiss cantons.'

   <country> Country </country> [0..1]
   'The ISO 3166 standard code for the country within which the postal address is located.'

   <postalCode> xsd:string </postalCode> [0..1]
   'The code, required for computerised mail sorting systems, that is allocated to a physical address by a national postal
   authority.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Address">
  <xsd:sequence>
    <xsd:element name="streetAddress" type=" StreetAddress " minOccurs="0"/>
    <xsd:element name="city" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="state" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="country" type=" Country " minOccurs="0"/>
    <xsd:element name="postalCode" type=" xsd:string " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableDate

[Table of contents]

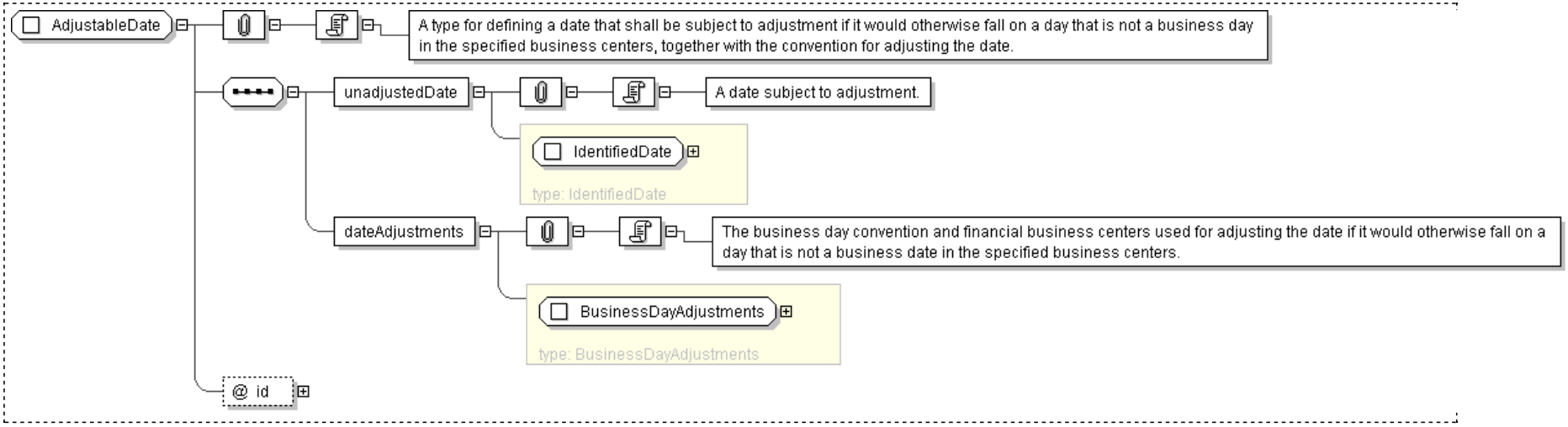
Super-types:	None
Sub-types:	None

Name	AdjustableDate
Used by (from the same schema document)	Complex Type AdjustableOrRelativeDate , Complex Type Payment
Abstract	no
Documentation	A type for defining a date that shall be subject to adjustment if it would otherwise fall on a day that is not a business day in the specified business centers, together with the convention for adjusting the date.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedDate> IdentifiedDate </unadjustedDate> [1]
    'A date subject to adjustment.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
    'The business day convention and financial business centers used for adjusting the date if it would otherwise fall on a day that is
    not a business date in the specified business centers.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDate">
   <xsd:sequence>
      <xsd:element name="unadjustedDate" type=" IdentifiedDate "/>
      <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableDate2

[Table of contents]

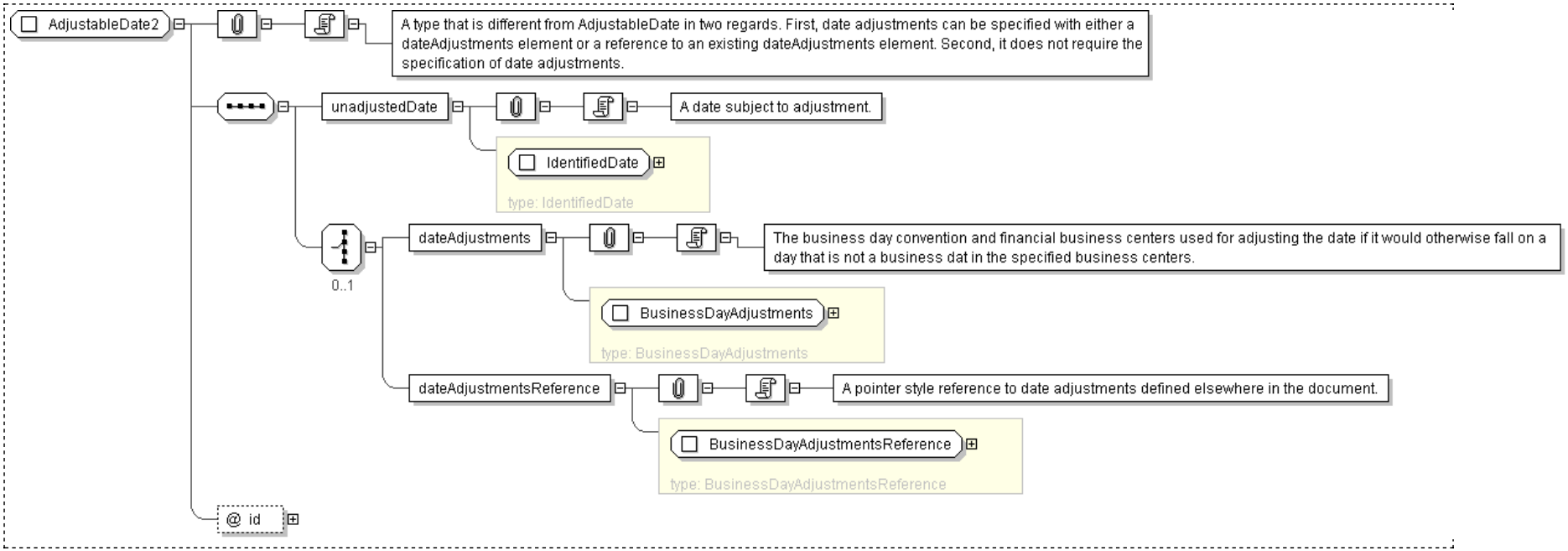
Super-types:	None
Sub-types:	None

Name	AdjustableDate2
Abstract	no
Documentation	A type that is different from AdjustableDate in two regards. First, date adjustments can be specified with either a dateAdjustments element or a reference to an existing dateAdjustments element. Second, it does not require the specification of date adjustments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedDate> IdentifiedDate </unadjustedDate> [1]
    'A date subject to adjustment.'

   Start Choice [0..1]
      <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
       'The business day convention and financial business centers used for adjusting the date if it would otherwise fall on a day that is not a
       business dat in the specified business centers.'

      <dateAdjustmentsReference> BusinessDayAdjustmentsReference </dateAdjustmentsReference> [1]
       'A pointer style reference to date adjustments defined elsewhere in the document.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDate2">
   <xsd:sequence>
      <xsd:element name="unadjustedDate" type=" IdentifiedDate "/>
      <xsd:choice minOccurs="0">
         <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments "/>
         <xsd:element name="dateAdjustmentsReference" type=" BusinessDayAdjustmentsReference "/>
      </xsd:choice>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableDates

[Table of contents]

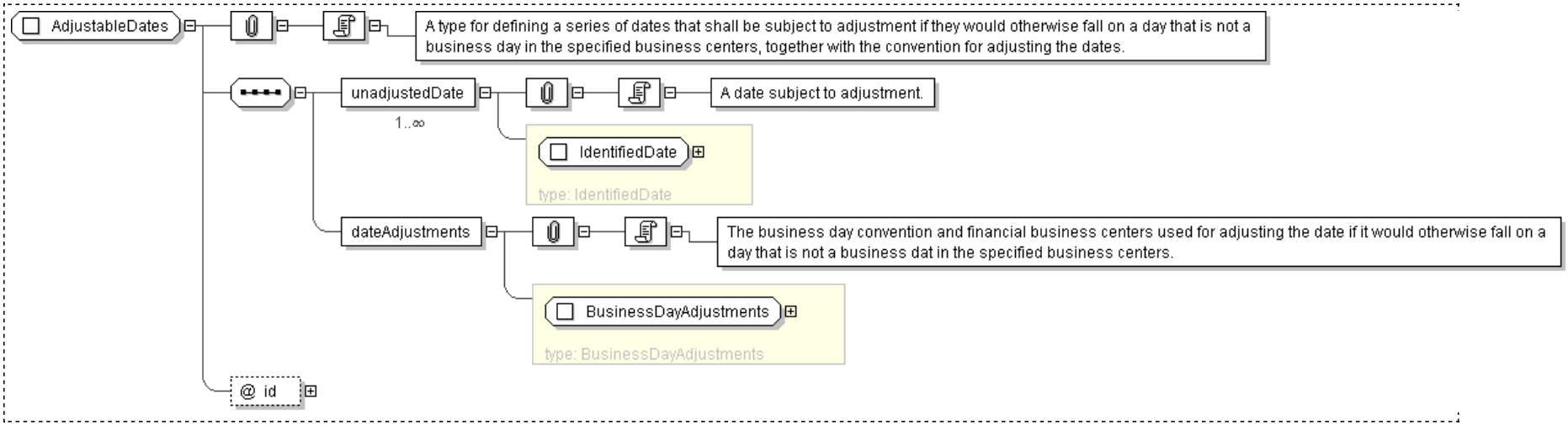
Super-types:	None
Sub-types:	None

Name	AdjustableDates
Used by (from the same schema document)	Complex Type AdjustableDatesOrRelativeDateOffset , Complex Type AdjustableOrRelativeDates , Complex Type AdjustableRelativeOrPeriodicDates , Complex Type AdjustableRelativeOrPeriodicDates2
Abstract	no
Documentation	A type for defining a series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified business centers, together with the convention for adjusting the dates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <unadjustedDate> IdentifiedDate </unadjustedDate> [1..*]
    'A date subject to adjustment.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
    'The business day convention and financial business centers used for adjusting the date if it would otherwise fall on a day that is
    not a business dat in the specified business centers.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDates">
  <xsd:sequence>
    <xsd:element name="unadjustedDate" type=" IdentifiedDate " maxOccurs="unbounded"/>
    <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableDatesOrRelativeDateOffset

[Table of contents]

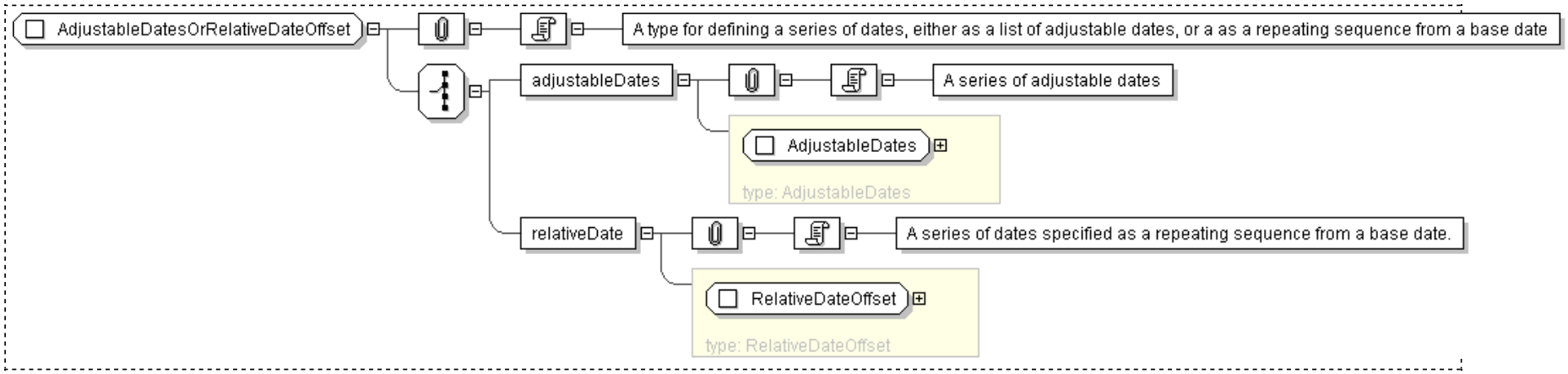
Super-types:	None
Sub-types:	None

Name	AdjustableDatesOrRelativeDateOffset
Abstract	no
Documentation	A type for defining a series of dates, either as a list of adjustable dates, or a as a repeating sequence from a base date

XML Instance Representation

```
<...>
  Start Choice [1]
    <adjustableDates> AdjustableDates </adjustableDates> [1]
    'A series of adjustable dates'

    <relativeDate> RelativeDateOffset </relativeDate> [1]
    'A series of dates specified as a repeating sequence from a base date.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDatesOrRelativeDateOffset">
  <xsd:choice>
    <xsd:element name="adjustableDates" type=" AdjustableDates "/>
    <xsd:element name="relativeDate" type=" RelativeDateOffset "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableOrRelativeAndAdjustedDate

[Table of contents]

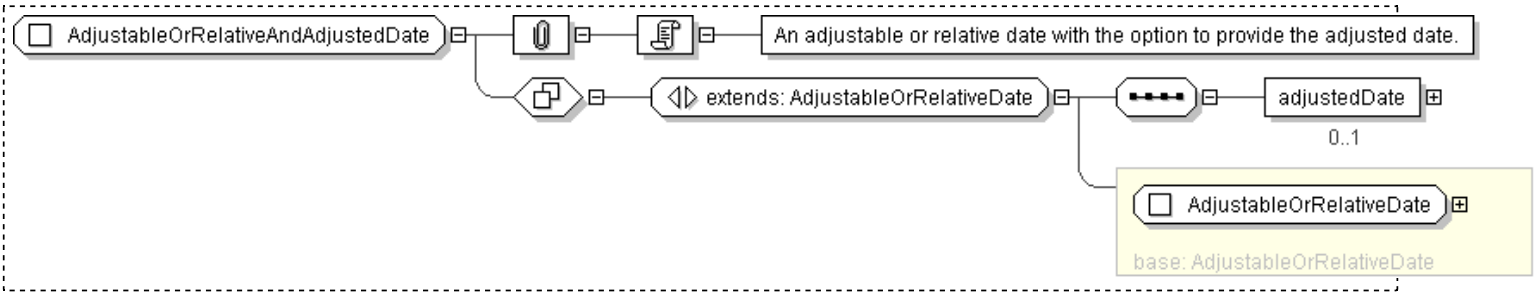
Super-types:	AdjustableOrRelativeDate < AdjustableOrRelativeAndAdjustedDate (by extension)
Sub-types:	None

Name	AdjustableOrRelativeAndAdjustedDate
Used by (from the same schema document)	Complex Type SimplePayment
Abstract	no
Documentation	An adjustable or relative date with the option to provide the adjusted date.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <adjustableDate> AdjustableDate </adjustableDate> [1]
      'A date that shall be subject to adjustment if it would otherwise fall on a day that is not
      a business day in the specified business centers, together with the convention for
      adjusting the date.'

      <relativeDate> RelativeDateOffset </relativeDate> [1]
      'A date specified as some offset to another date (the anchor date).'

   End Choice
   <adjustedDate> IdentifiedDate </adjustedDate> [0..1]
   'The adjusted date. This date should already be adjusted for any applicable business day
   convention. This component is not intended for use in trade confirmation but my be specified
   to allow the fee structure to also serve as a cashflow type component (all dates the the
   Cashflows type are adjusted payment dates).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableOrRelativeAndAdjustedDate">
  <xsd:complexContent>
    <xsd:extension base=" AdjustableOrRelativeDate ">
      <xsd:sequence>
        <xsd:element name="adjustedDate" type=" IdentifiedDate " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableOrRelativeDate

[Table of contents]

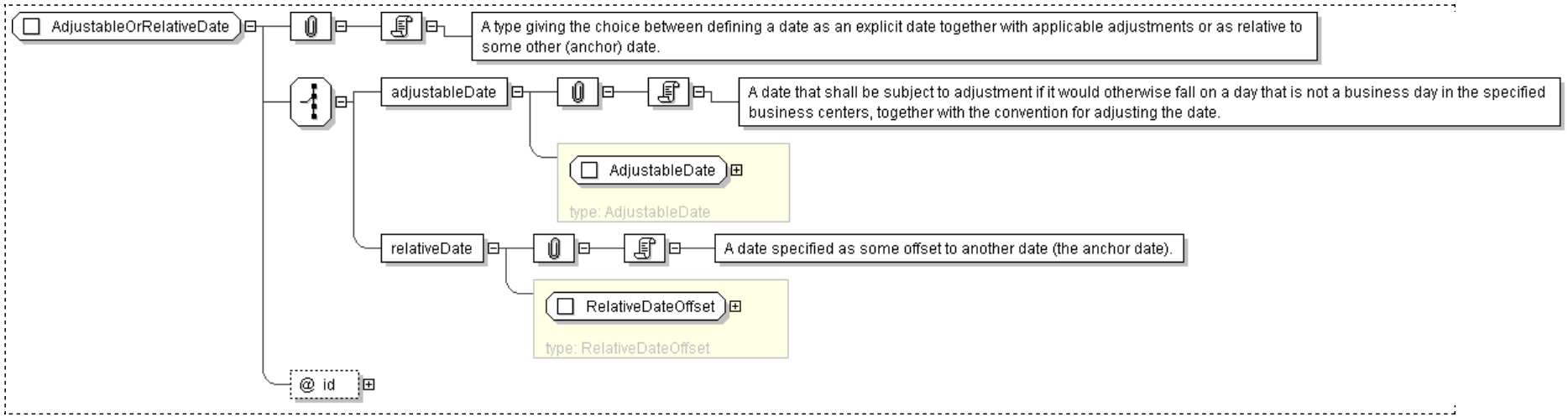
Super-types:	None
Sub-types:	• AdjustableOrRelativeAndAdjustedDate (by extension)

Name	AdjustableOrRelativeDate
Used by (from the same schema document)	Complex Type AmericanExercise , Complex Type AmericanExercise , Complex Type EuropeanExercise , Complex Type PaymentBaseExtended , Complex Type PeriodicDates , Complex Type PeriodicDates , Complex Type SharedAmericanExercise , Complex Type SharedAmericanExercise , Complex Type Stub , Complex Type Stub
Abstract	no
Documentation	A type giving the choice between defining a date as an explicit date together with applicable adjustments or as relative to some other (anchor) date.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
  Start Choice [1]
     <adjustableDate> AdjustableDate </adjustableDate> [1]
     'A date that shall be subject to adjustment if it would otherwise fall on a day that is not a business day in the specified business
     centers, together with the convention for adjusting the date.'

     <relativeDate> RelativeDateOffset </relativeDate> [1]
     'A date specified as some offset to another date (the anchor date).'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableOrRelativeDate">
  <xsd:choice>
    <xsd:element name="adjustableDate" type=" AdjustableDate "/>
    <xsd:element name="relativeDate" type=" RelativeDateOffset "/>
```

```
   </xsd:choice>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableOrRelativeDates

[Table of contents]

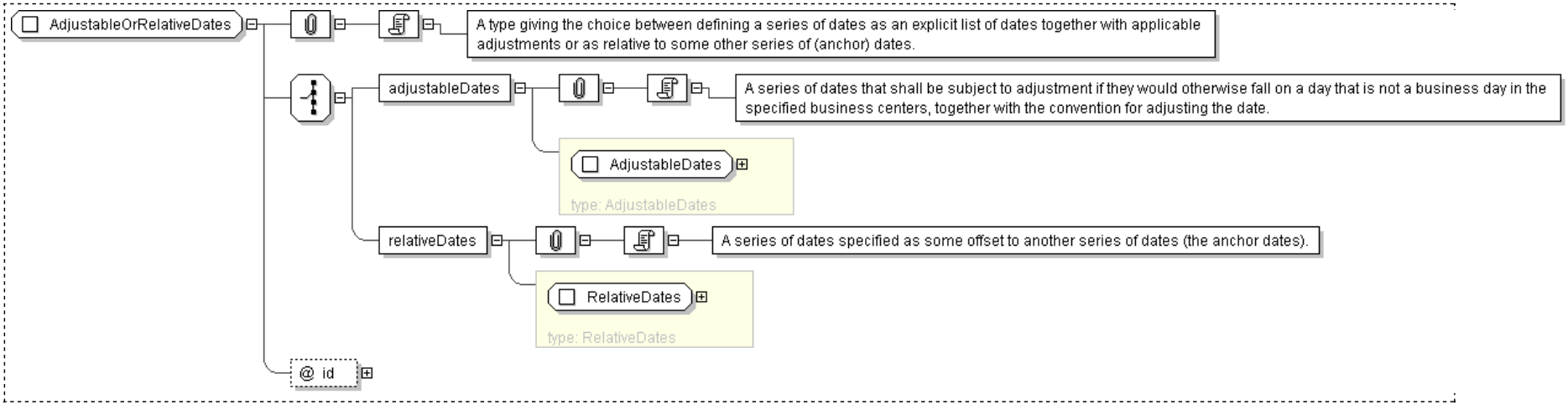
Super-types:	None
Sub-types:	None

Name	AdjustableOrRelativeDates
Used by (from the same schema document)	Complex Type AmericanExercise , Complex Type BermudaExercise , Complex Type BermudaExercise , Complex Type EuropeanExercise
Abstract	no
Documentation	A type giving the choice between defining a series of dates as an explicit list of dates together with applicable adjustments or as relative to some other series of (anchor) dates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
  Start Choice [1]
     <adjustableDates> AdjustableDates </adjustableDates> [1]
     'A series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified
     business centers, together with the convention for adjusting the date.'

     <relativeDates> RelativeDates </relativeDates> [1]
     'A series of dates specified as some offset to another series of dates (the anchor dates).'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableOrRelativeDates">
  <xsd:choice>
    <xsd:element name="adjustableDates" type=" AdjustableDates "/>
    <xsd:element name="relativeDates" type=" RelativeDates "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableRelativeOrPeriodicDates

[Table of contents]

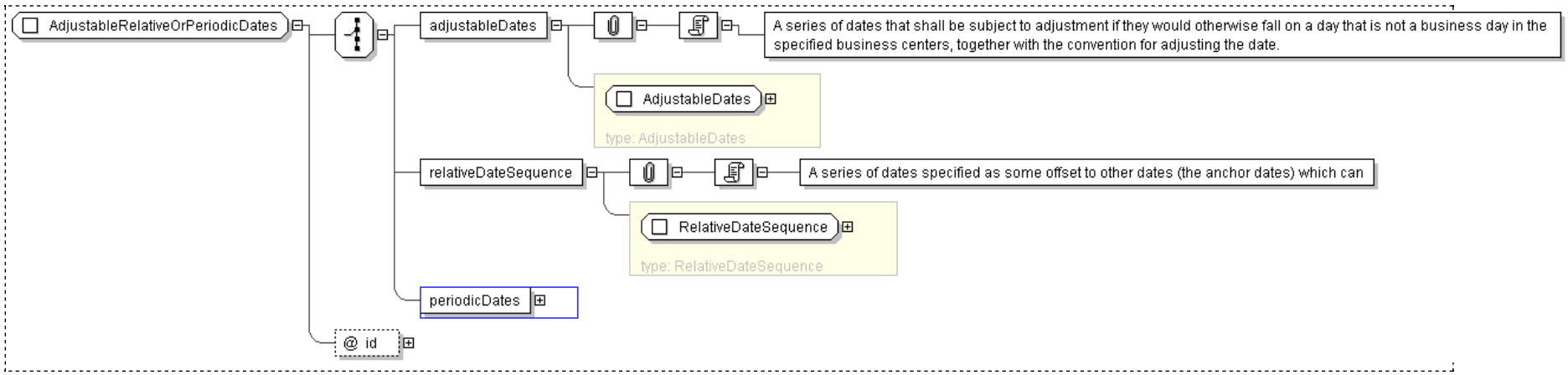
Super-types:	None
Sub-types:	None

Name	AdjustableRelativeOrPeriodicDates
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <adjustableDates> AdjustableDates </adjustableDates> [1]
      'A series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified
      business centers, together with the convention for adjusting the date.'

      <relativeDateSequence> RelativeDateSequence </relativeDateSequence> [1]
      'A series of dates specified as some offset to other dates (the anchor dates) which can'

      <periodicDates> PeriodicDates </periodicDates> [1]
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableRelativeOrPeriodicDates">
  <xsd:choice>
    <xsd:element name="adjustableDates" type=" AdjustableDates "/>
    <xsd:element name="relativeDateSequence" type=" RelativeDateSequence "/>
    <xsd:element name="periodicDates" type=" PeriodicDates "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustableRelativeOrPeriodicDates2

[Table of contents]

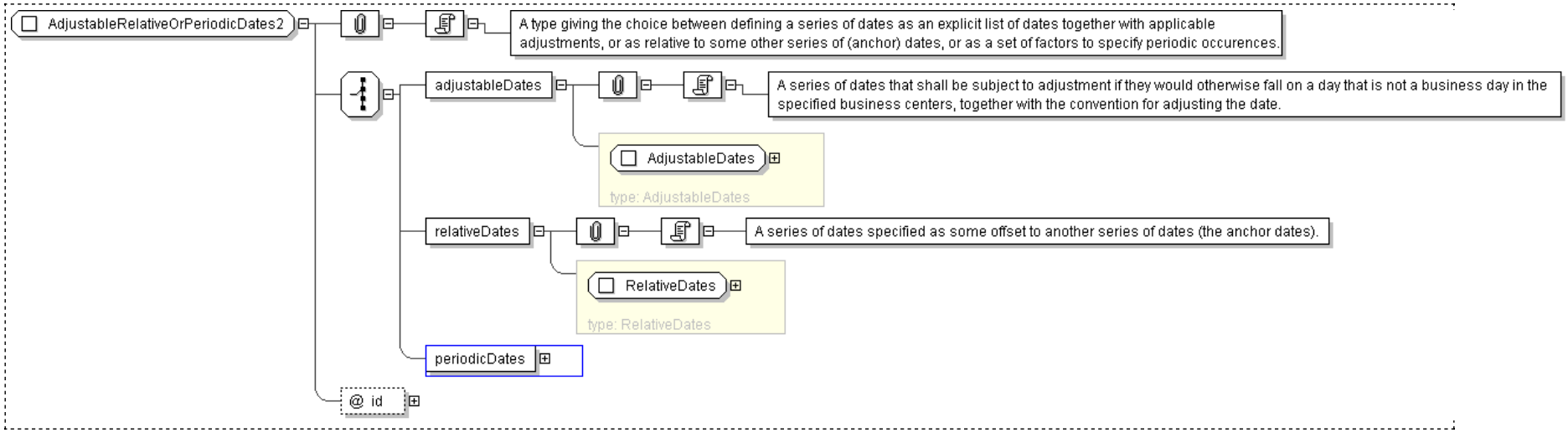
Super-types:	None
Sub-types:	None

Name	AdjustableRelativeOrPeriodicDates2
Abstract	no
Documentation	A type giving the choice between defining a series of dates as an explicit list of dates together with applicable adjustments, or as relative to some other series of (anchor) dates, or as a set of factors to specify periodic occurences.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <adjustableDates> AdjustableDates </adjustableDates> [1]
      'A series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified
      business centers, together with the convention for adjusting the date.'

      <relativeDates> RelativeDates </relativeDates> [1]
      'A series of dates specified as some offset to another series of dates (the anchor dates).'

      <periodicDates> PeriodicDates </periodicDates> [1]
   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableRelativeOrPeriodicDates2">
  <xsd:choice>
    <xsd:element name="adjustableDates" type=" AdjustableDates "/>
    <xsd:element name="relativeDates" type=" RelativeDates "/>
    <xsd:element name="periodicDates" type=" PeriodicDates "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AdjustedRelativeDateOffset

[Table of contents]

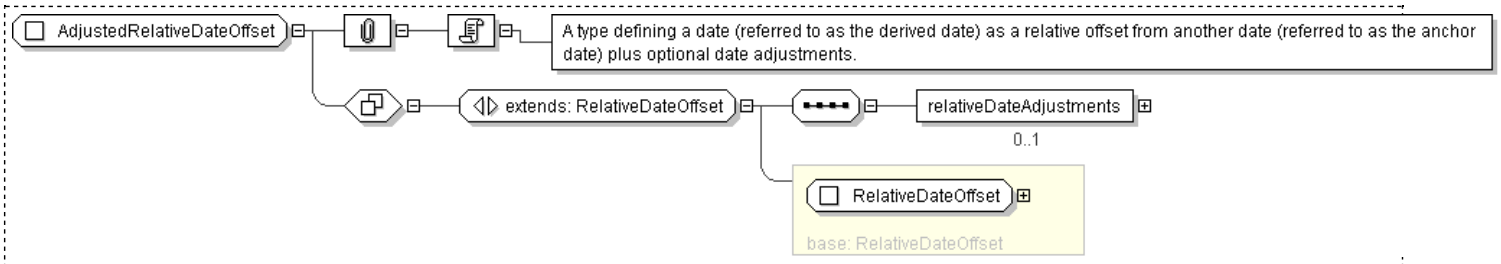
Super-types:	Period < Offset (by extension) < RelativeDateOffset (by extension) < AdjustedRelativeDateOffset (by extension)
Sub-types:	None

Name	AdjustedRelativeDateOffset
Abstract	no
Documentation	A type defining a date (referred to as the derived date) as a relative offset from another date (referred to as the anchor date) plus optional date adjustments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to
   another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0 (zero) then
   period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
   'In the case of an offset specified as a number of days, this element defines whether consideration is given as to
   whether a day is a good business day or not. If a day type of business days is specified then non-business days
   are ignored when calculating the offset. The financial business centers to use for determination of business days
   are implied by the context in which this element is used. This element must only be included when the offset is
   specified as a number of days. If the offset is zero days then the dayType element should not be included.'

   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
   'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
         'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set
         of business centers is used to determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
   <dateRelativeTo> DateReference </dateRelativeTo> [1]
   'Specifies the anchor as an href attribute. The href attribute value is a pointer style reference to the element
   or component elsewhere in the document where the anchor date is defined.'

   <relativeDateAdjustments> BusinessDayAdjustments </relativeDateAdjustments> [0..1]
   'The business day convention and financial business centers used for adjusting the relative date if it would
   otherwise fall on a day that is not a business date in the specified business centers.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustedRelativeDateOffset">
  <xsd:complexContent>
    <xsd:extension base=" RelativeDateOffset ">
      <xsd:sequence>
        <xsd:element name="relativeDateAdjustments" type=" BusinessDayAdjustments " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AmericanExercise

[Table of contents]

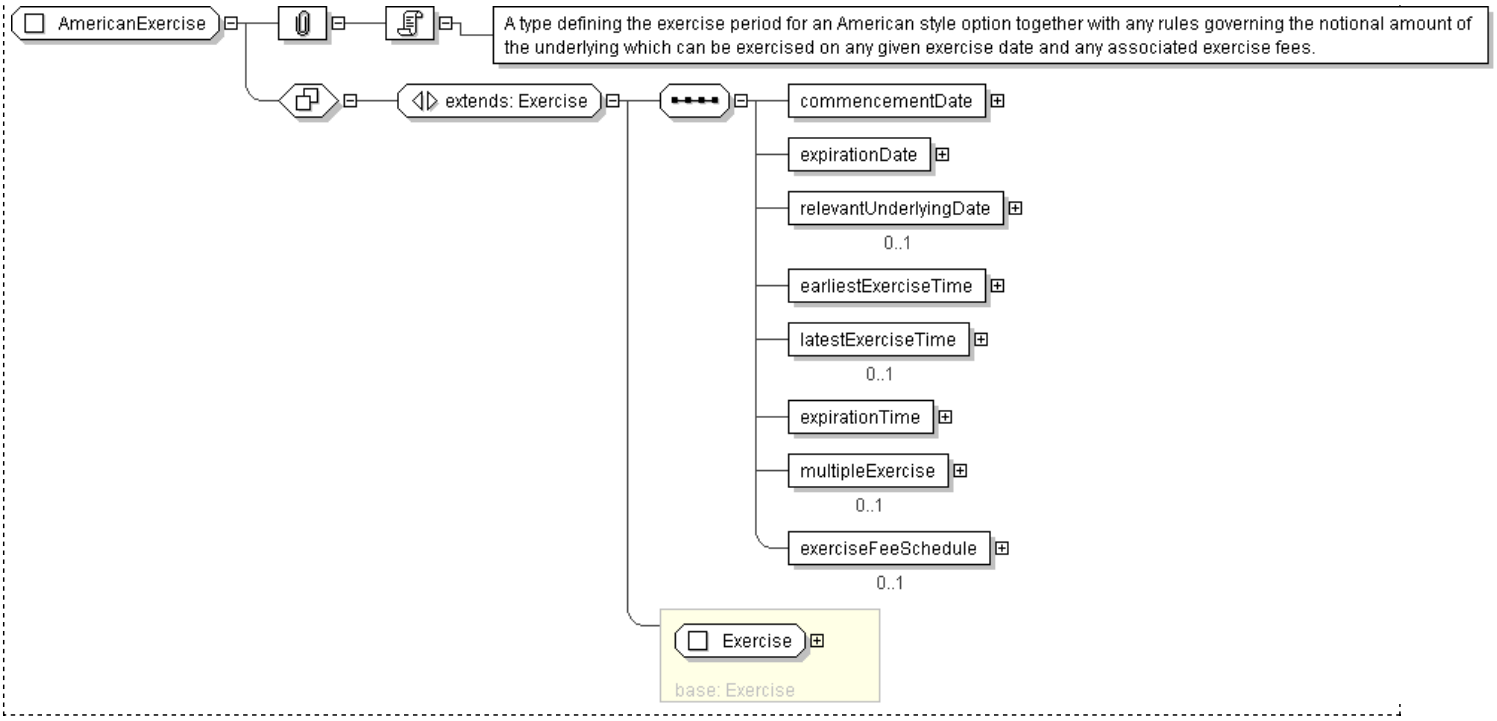
Super-types:	Exercise < AmericanExercise (by extension)
Sub-types:	None

Name	AmericanExercise
Used by (from the same schema document)	Element americanExercise
Abstract	no
Documentation	A type defining the exercise period for an American style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
     'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
     'The last day within an exercise period for an American style option. For a European style option it is the
     only day within the exercise period.'

   <relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]
     'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g.
     in a swaption it is the effective date, in an extendible/cancelable provision it is the termination date).'

   <earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]
     'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent) i)
     on the expriation date, in the case of a European style option, (ii) on each bermuda option exercise date and
     the expiration date, in the case of a Bermuda style option the commencement date to, and including, the
     expiration date , in the case of an American option.'

   <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [0..1]
     'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration
     date) within the exercise period that notice can be given by the buyer to the seller or seller\'s agent.
     Notice of exercise given after this time will be deemed to have been given on the next exercise business
     day.'

   <expirationTime> BusinessCenterTime </expirationTime> [1]
     'The latest time for exercise on expirationDate.'

   <multipleExercise> MultipleExercise </multipleExercise> [0..1]
     'As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the
     right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more
     days in the exercise period, but on any such day may not exercise less than the minimum notional amount or
     more that the maximum notional amount, and if an integral multiple amount is specified, the notional amount
     exercised must be equal to, or be an intergral multiple of, the integral multiple amount.'

   <exerciseFeeSchedule> ExerciseFeeSchedule </exerciseFeeSchedule> [0..1]
     'The fees associated with an exercise date. The fees are conditional on the exercise occuring. The fees can be
     specified as actual currency amounts or as percentages of the notional amount being exercised.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AmericanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:element name="commencementDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="relevantUnderlyingDate" type=" AdjustableOrRelativeDates " minOccurs="0"/>
        <xsd:element name="earliestExerciseTime" type=" BusinessCenterTime "/>
        <xsd:element name="latestExerciseTime" type=" BusinessCenterTime " minOccurs="0"/>
        <xsd:element name="expirationTime" type=" BusinessCenterTime "/>
        <xsd:element name="multipleExercise" type=" MultipleExercise " minOccurs="0"/>
        <xsd:element name="exerciseFeeSchedule" type=" ExerciseFeeSchedule " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AmountReference

[Table of contents]

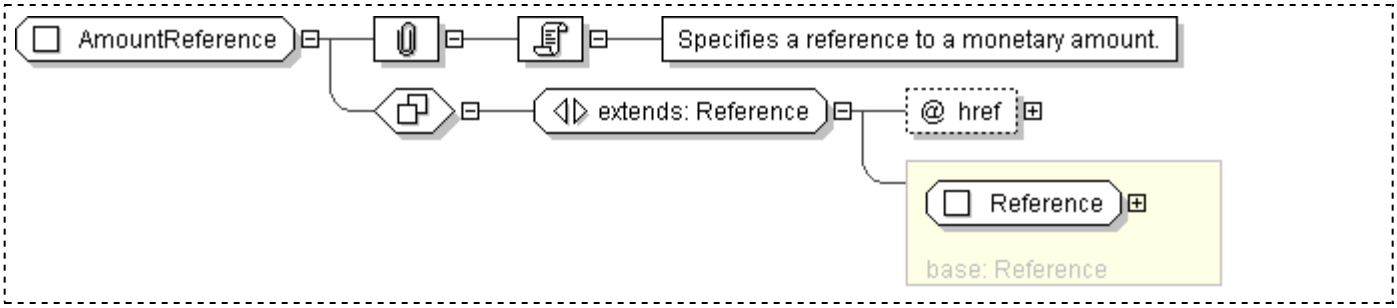
Super-types:	Reference < AmountReference (by extension)
Sub-types:	None

Name	AmountReference
Abstract	no
Documentation	Specifies a reference to a monetary amount.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AmountReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AmountSchedule

[Table of contents]

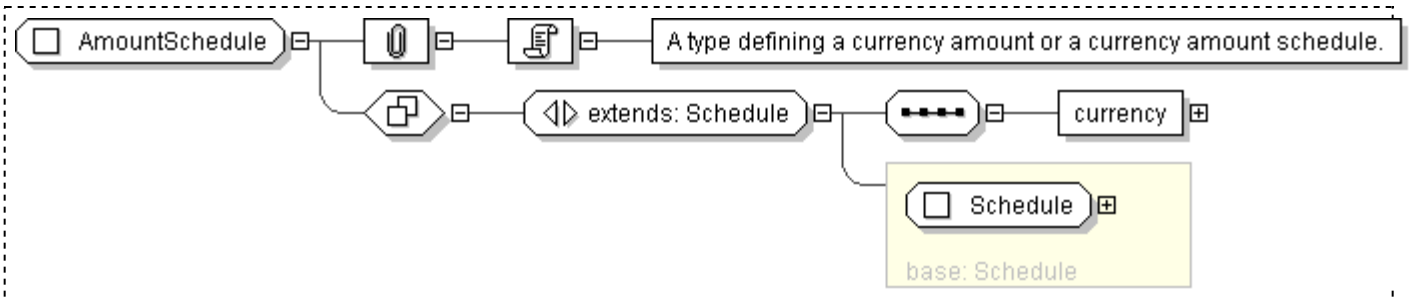
Super-types:	Schedule < AmountSchedule (by extension)
Sub-types:	None

Name	AmountSchedule
Used by (from the same schema document)	Complex Type ExerciseFeeSchedule
Abstract	no
Documentation	A type defining a currency amount or a currency amount schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> xsd:decimal </initialValue> [1]
    'The initial rate or amount, as the case may be. An initial rate of 5% would be
    represented as 0.05.'

   <step> Step </step> [0..*]
    'The schedule of step date and value pairs. On each step date the associated
    step value becomes effective A list of steps may be ordered in the document by
    ascending step date. An FpML document containing an unordered list of steps is
    still regarded as a conformant document.'

   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AmountSchedule">
   <xsd:complexContent>
      <xsd:extension base=" Schedule ">
         <xsd:sequence>
            <xsd:element name="currency" type=" Currency "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AutomaticExercise

[Table of contents]

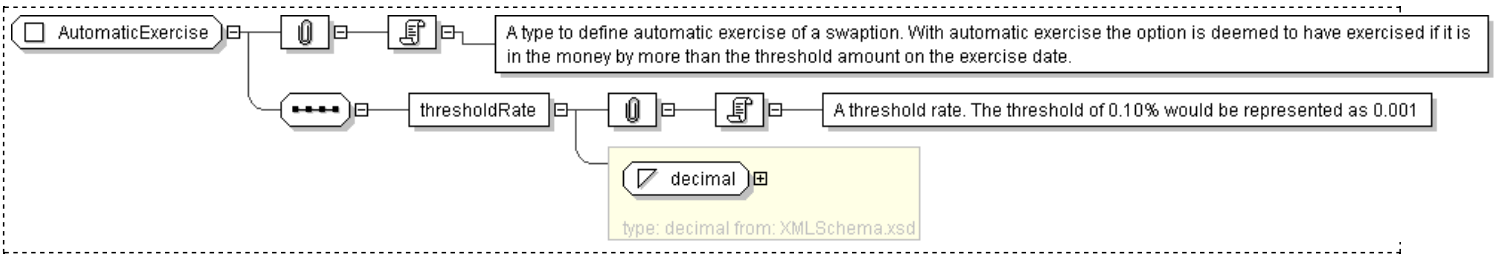
Super-types:	None
Sub-types:	None

Name	AutomaticExercise
Used by (from the same schema document)	Complex Type ExerciseProcedure
Abstract	no
Documentation	A type to define automatic exercise of a swaption. With automatic exercise the option is deemed to have exercised if it is in the money by more than the threshold amount on the exercise date.

XML Instance Representation

```
<...>
  <thresholdRate> xsd:decimal </thresholdRate> [1]
  'A threshold rate. The threshold of 0.10% would be represented as 0.001'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AutomaticExercise">
  <xsd:sequence>
    <xsd:element name="thresholdRate" type=" xsd:decimal "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AverageDailyTradingVolumeLimit

[Table of contents]

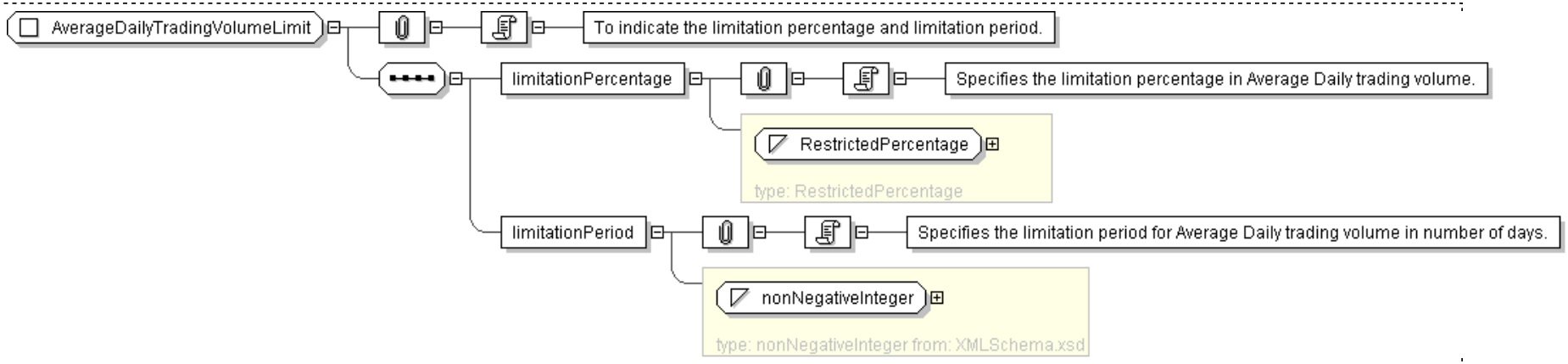
Super-types:	None
Sub-types:	None

Name	AverageDailyTradingVolumeLimit
Abstract	no
Documentation	To indicate the limitation percentage and limitation period.

XML Instance Representation

```
<...>
  <limitationPercentage> RestrictedPercentage </limitationPercentage> [1]
   'Specifies the limitation percentage in Average Daily trading volume.'

  <limitationPeriod> xsd:nonNegativeInteger </limitationPeriod> [1]
   'Specifies the limitation period for Average Daily trading volume in number of days.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AverageDailyTradingVolumeLimit">
  <xsd:sequence>
    <xsd:element name="limitationPercentage" type=" RestrictedPercentage "/>
    <xsd:element name="limitationPeriod" type=" xsd:nonNegativeInteger "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Beneficiary

[Table of contents]

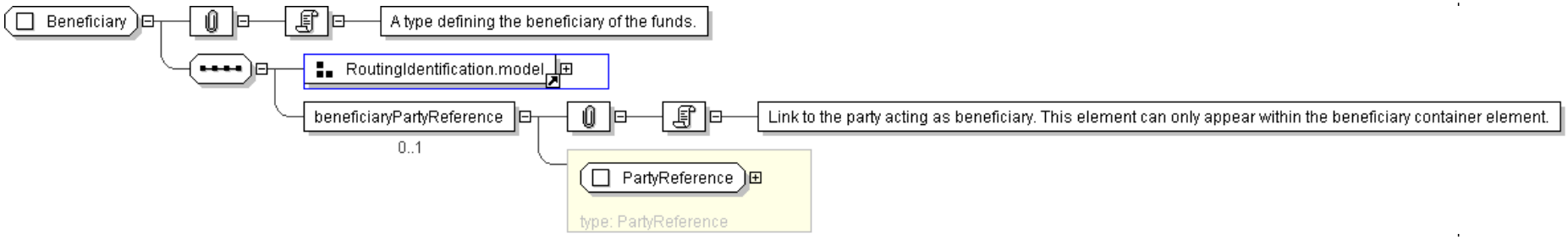
Super-types:	None
Sub-types:	None

Name	Beneficiary
Used by (from the same schema document)	Complex Type SettlementInstruction , Complex Type SettlementInstruction
Abstract	no
Documentation	A type defining the beneficiary of the funds.

XML Instance Representation

```
<...>
  Start Choice [1]
    <routingIds> RoutingIds </routingIds> [1]
     'A set of unique identifiers for a party, eachone identifying the party within a payment system. The assumption is that each
     party will not have more than one identifier within the same payment system.'

    <routingExplicitDetails> RoutingExplicitDetails </routingExplicitDetails> [1]
     'A set of details that is used to identify a party involved in the routing of a payment when the party does not have a code that
     identifies it within one of the recognized payment systems.'

    <routingIdsAndExplicitDetails> RoutingIdsAndExplicitDetails </routingIdsAndExplicitDetails> [1]
     'A combination of coded payment system identifiers and details for physical addressing for a party involved in the routing of a
     payment.'

  End Choice
  <beneficiaryPartyReference> PartyReference </beneficiaryPartyReference> [0..1]
  'Link to the party acting as beneficiary. This element can only appear within the beneficiary container element.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Beneficiary">
  <xsd:sequence>
    <xsd:group ref=" RoutingIdentification.model "/>
    <xsd:element name="beneficiaryPartyReference" type=" PartyReference " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BermudaExercise

[Table of contents]

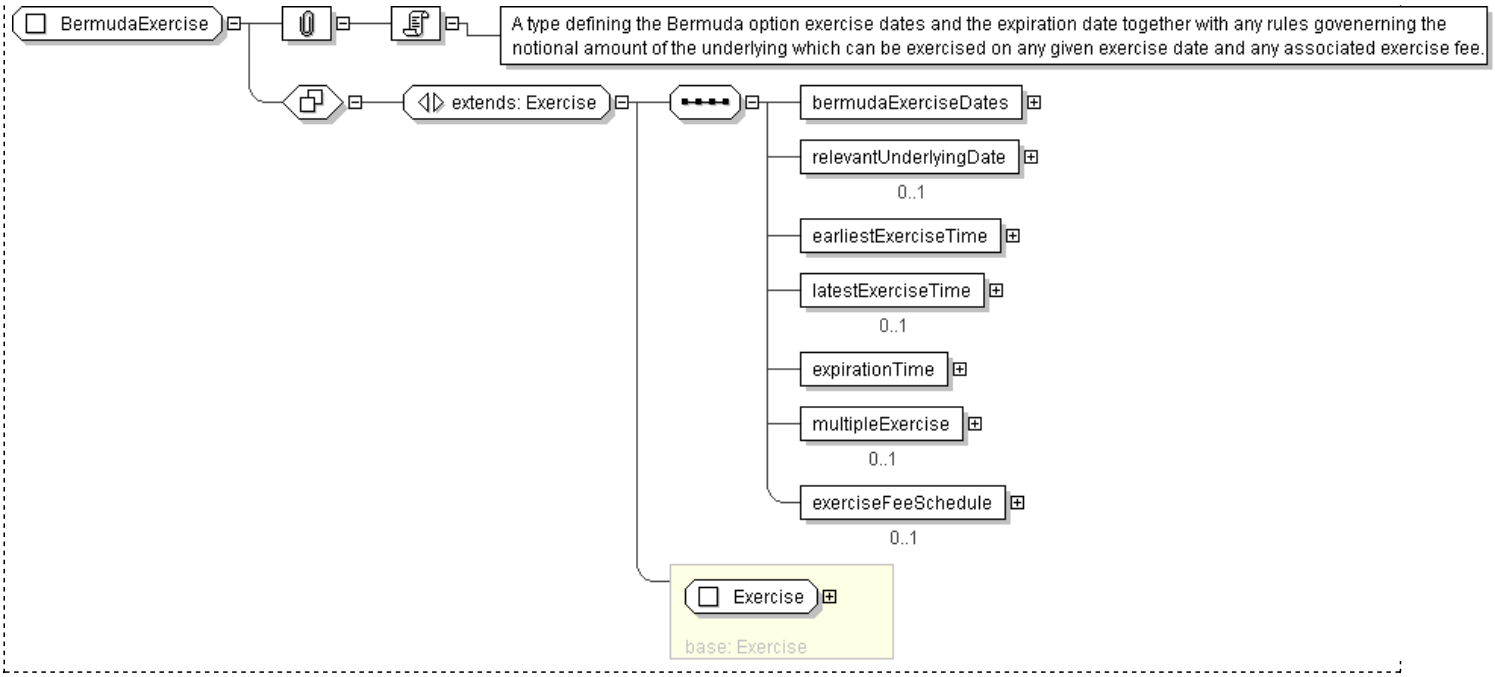
Super-types:	Exercise < BermudaExercise (by extension)
Sub-types:	None

Name	BermudaExercise
Used by (from the same schema document)	Element bermudaExercise
Abstract	no
Documentation	A type defining the Bermuda option exercise dates and the expiration date together with any rules govenerning the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fee.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <bermudaExerciseDates> AdjustableOrRelativeDates </bermudaExerciseDates> [1]
   'The dates the define the Bermuda option exercise dates and the expiration date. The last specified date is
   assumed to be the expiration date. The dates can either be specified as a series of explicit dates and
   associated adjustments or as a series of dates defined relative to another schedule of dates, for example,
   the calculation period start dates. Where a relative series of dates are defined the first and last possible
   exercise dates can be separately specified.'

   <relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]
   'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g.
   in a swaption it is the effective date, in an extendible/cancelable provision it is the termination date).'

   <earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]
   'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent)
   i) on the expriation date, in the case of a European style option, (ii) on each bermuda option exercise date
   and the expiration date, in the case of a Bermuda style option the commencement date to, and including, the
   expiration date , in the case of an American option.'

   <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [0..1]
   'For a Bermuda or American style option, the latest time on an exercise business day (excluding the
   expiration date) within the exercise period that notice can be given by the buyer to the seller or seller\'s
   agent. Notice of exercise given after this time will be deemed to have been given on the next exercise
   business day.'

   <expirationTime> BusinessCenterTime </expirationTime> [1]
   'The latest time for exercise on expirationDate.'

   <multipleExercise> MultipleExercise </multipleExercise> [0..1]
   'As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the
   right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more
   days in the exercise period, but on any such day may not exercise less than the minimum notional amount or
   more that the maximum notional amount, and if an integral multiple amount is specified, the notional amount
   exercised must be equal to, or be an intergral multiple of, the integral multiple amount.'

   <exerciseFeeSchedule> ExerciseFeeSchedule </exerciseFeeSchedule> [0..1]
   'The fees associated with an exercise date. The fees are conditional on the exercise occuring. The fees can
   be specified as actual currency amounts or as percentages of the notional amount being exercised.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BermudaExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:element name="bermudaExerciseDates" type=" AdjustableOrRelativeDates "/>
        <xsd:element name="relevantUnderlyingDate" type=" AdjustableOrRelativeDates " minOccurs="0"/>
        <xsd:element name="earliestExerciseTime" type=" BusinessCenterTime "/>
        <xsd:element name="latestExerciseTime" type=" BusinessCenterTime " minOccurs="0"/>
        <xsd:element name="expirationTime" type=" BusinessCenterTime "/>
        <xsd:element name="multipleExercise" type=" MultipleExercise " minOccurs="0"/>
        <xsd:element name="exerciseFeeSchedule" type=" ExerciseFeeSchedule " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BrokerConfirmation

[Table of contents]

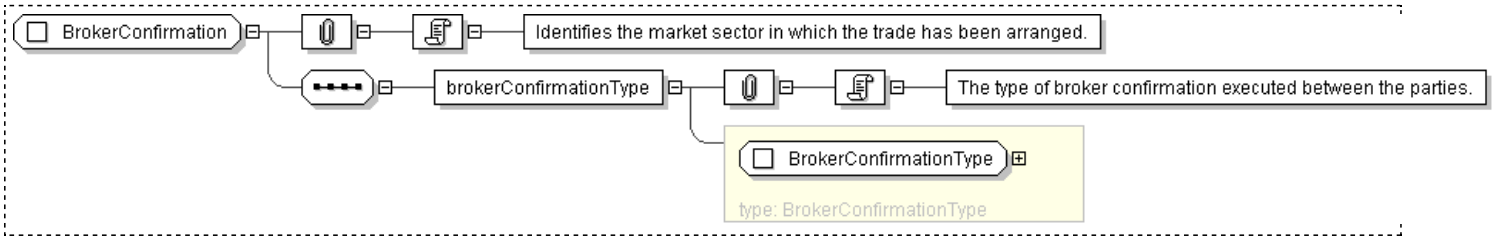
Super-types:	None
Sub-types:	None

Name	BrokerConfirmation
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	Identifies the market sector in which the trade has been arranged.

XML Instance Representation

```
<...>
  <brokerConfirmationType> BrokerConfirmationType </brokerConfirmationType> [1]
  'The type of broker confirmation executed between the parties.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BrokerConfirmation">
  <xsd:sequence>
    <xsd:element name="brokerConfirmationType" type=" BrokerConfirmationType "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BrokerConfirmationType

[Table of contents]

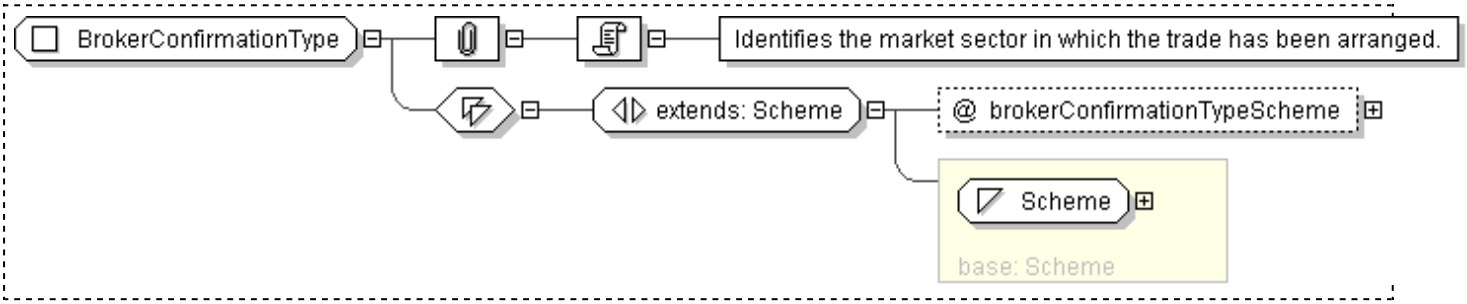
Super-types:	xsd:normalizedString < Scheme (by restriction) < BrokerConfirmationType (by extension)
Sub-types:	None

Name	BrokerConfirmationType
Used by (from the same schema document)	Complex Type BrokerConfirmation
Abstract	no
Documentation	Identifies the market sector in which the trade has been arranged.

XML Instance Representation

```
<...
 brokerConfirmationTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BrokerConfirmationType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="brokerConfirmationTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/broker-confirmation-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessCenter

[Table of contents]

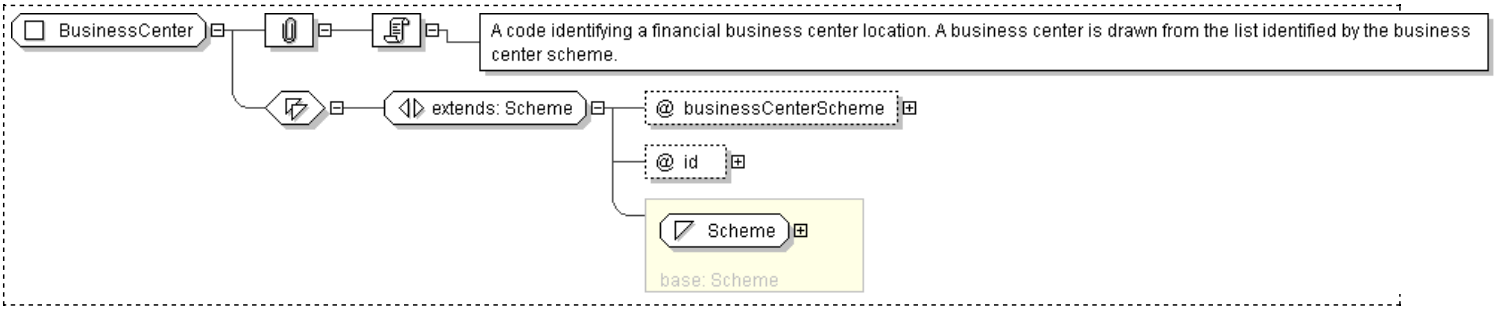
Super-types:	xsd:normalizedString < Scheme (by restriction) < BusinessCenter (by extension)
Sub-types:	None

Name	BusinessCenter
Used by (from the same schema document)	Complex Type BusinessCenters , Complex Type BusinessCenterTime , Complex Type ExerciseNotice
Abstract	no
Documentation	A code identifying a financial business center location. A business center is drawn from the list identified by the business center scheme.

XML Instance Representation

```
<...
 businessCenterScheme=" xsd:anyURI [0..1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessCenter">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="businessCenterScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/business-center"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessCenters

[Table of contents]

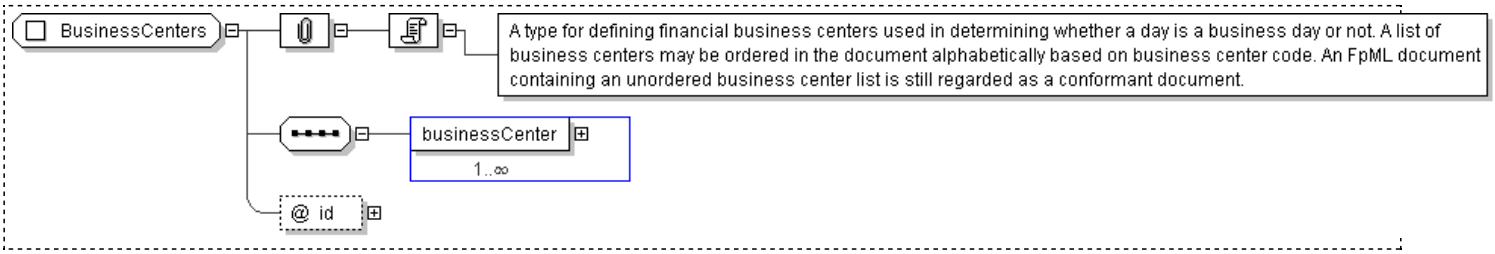
Super-types:	None
Sub-types:	None

Name	BusinessCenters
Used by (from the same schema document)	Model Group BusinessCentersOrReference.model
Abstract	no
Documentation	A type for defining financial business centers used in determining whether a day is a business day or not. A list of business centers may be ordered in the document alphabetically based on business center code. An FpML document containing an unordered business center list is still regarded as a conformant document.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <businessCenter> BusinessCenter </businessCenter> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessCenters">
  <xsd:sequence>
     <xsd:element name="businessCenter" type=" BusinessCenter " maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessCentersReference

[Table of contents]

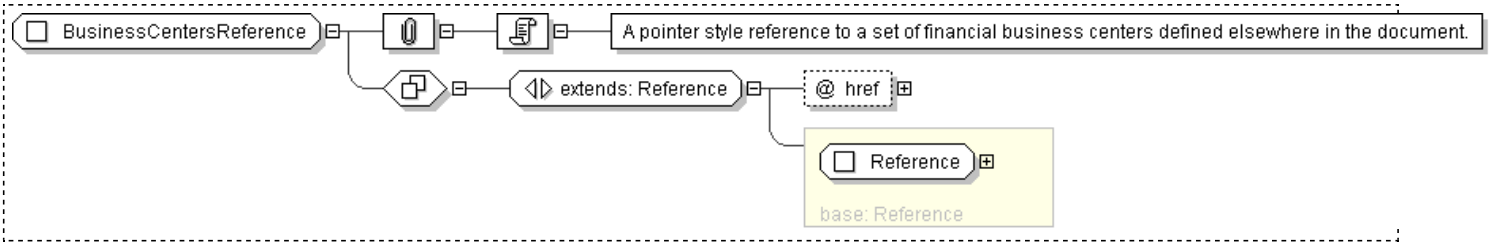
Super-types:	Reference < BusinessCentersReference (by extension)
Sub-types:	None

Name	BusinessCentersReference
Used by (from the same schema document)	Model Group BusinessCentersOrReference.model
Abstract	no
Documentation	A pointer style reference to a set of financial business centers defined elsewhere in the document.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessCentersReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="BusinessCenters"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessCenterTime

[Table of contents]

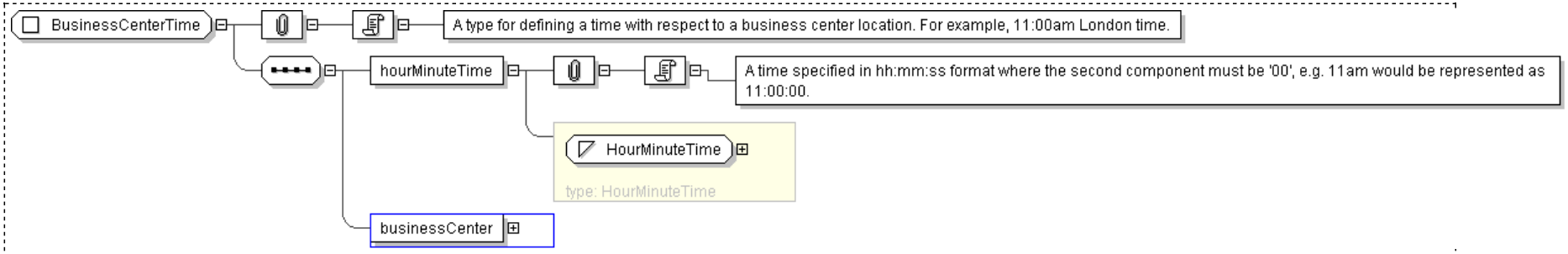
Super-types:	None
Sub-types:	None

Name	BusinessCenterTime
Used by (from the same schema document)	Complex Type AmericanExercise , Complex Type AmericanExercise , Complex Type AmericanExercise , Complex Type BermudaExercise , Complex Type BermudaExercise , Complex Type BermudaExercise , Complex Type EuropeanExercise , Complex Type EuropeanExercise , Complex Type FxSpotRateSource , Complex Type SharedAmericanExercise
Abstract	no
Documentation	A type for defining a time with respect to a business center location. For example, 11:00am London time.

XML Instance Representation

```
<...>
   <hourMinuteTime> HourMinuteTime </hourMinuteTime> [1]
   'A time specified in hh:mm:ss format where the second component must be \'00\', e.g. 11am would be represented as 11:00:00.'

   <businessCenter> BusinessCenter </businessCenter> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessCenterTime">
   <xsd:sequence>
      <xsd:element name="hourMinuteTime" type=" HourMinuteTime "/>
      <xsd:element name="businessCenter" type=" BusinessCenter "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessDateRange

[Table of contents]

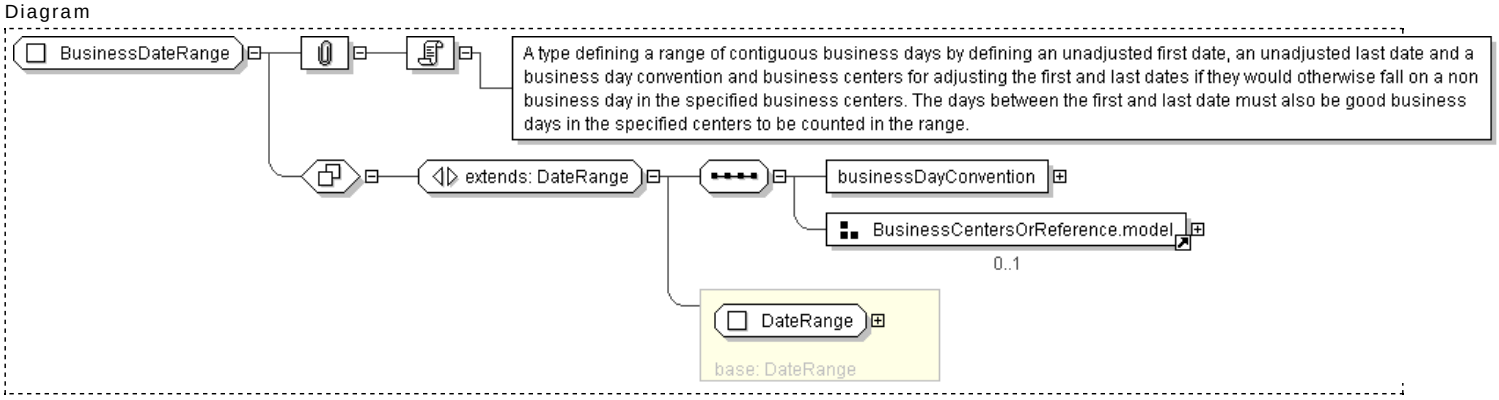
Super-types:	DateRange < BusinessDateRange (by extension)
Sub-types:	None

Name	BusinessDateRange
Abstract	no
Documentation	A type defining a range of contiguous business days by defining an unadjusted first date, an unadjusted last date and a business day convention and business centers for adjusting the first and last dates if they would otherwise fall on a non business day in the specified business centers. The days between the first and last date must also be good business days in the specified centers to be counted in the range.

XML Instance Representation

```
<...>
  <unadjustedFirstDate> xsd:date </unadjustedFirstDate> [1]
  'The first date of a date range.'

  <unadjustedLastDate> xsd:date </unadjustedLastDate> [1]
  'The last date of a date range.'

  <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
  'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

  Start Group: BusinessCentersOrReference.model [0..1]
    Start Choice [1]
      <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
      'A pointer style reference to a set of financial business centers defined elsewhere in the document. This
      set of business centers is used to determine whether a particular day is a business day or not.'

      <businessCenters> BusinessCenters </businessCenters> [1]
    End Choice
  End Group: BusinessCentersOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessDateRange">
  <xsd:complexContent>
    <xsd:extension base=" DateRange ">
      <xsd:sequence>
        <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
        <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessDayAdjustments

[Table of contents]

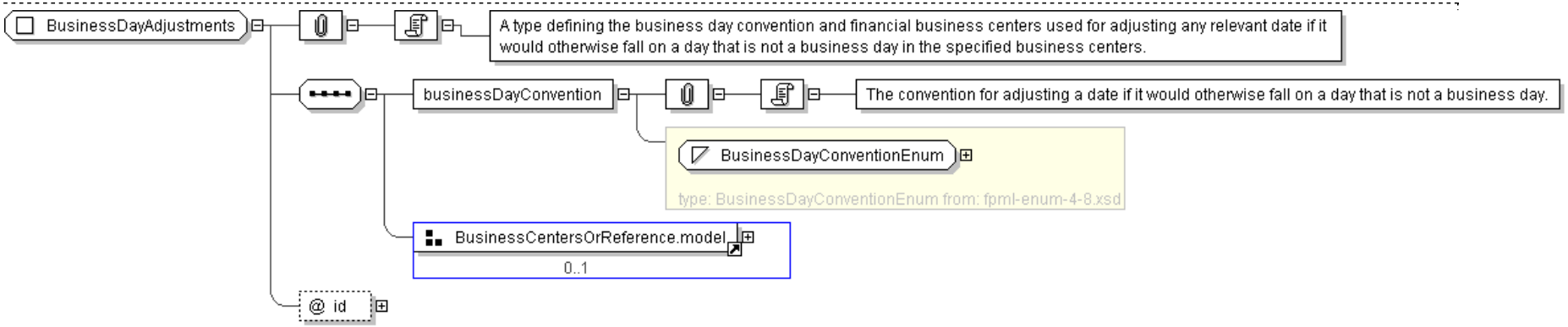
Super-types:	None
Sub-types:	None

Name	BusinessDayAdjustments
Used by (from the same schema document)	Complex Type AdjustableDate , Complex Type AdjustableDate2 , Complex Type AdjustableDates , Complex Type AdjustedRelativeDateOffset , Complex Type PeriodicDates
Abstract	no
Documentation	A type defining the business day convention and financial business centers used for adjusting any relevant date if it would otherwise fall on a day that is not a business day in the specified business centers.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
    'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
          'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business
          centers is used to determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessDayAdjustments">
   <xsd:sequence>
      <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
      <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: BusinessDayAdjustmentsReference

[Table of contents]

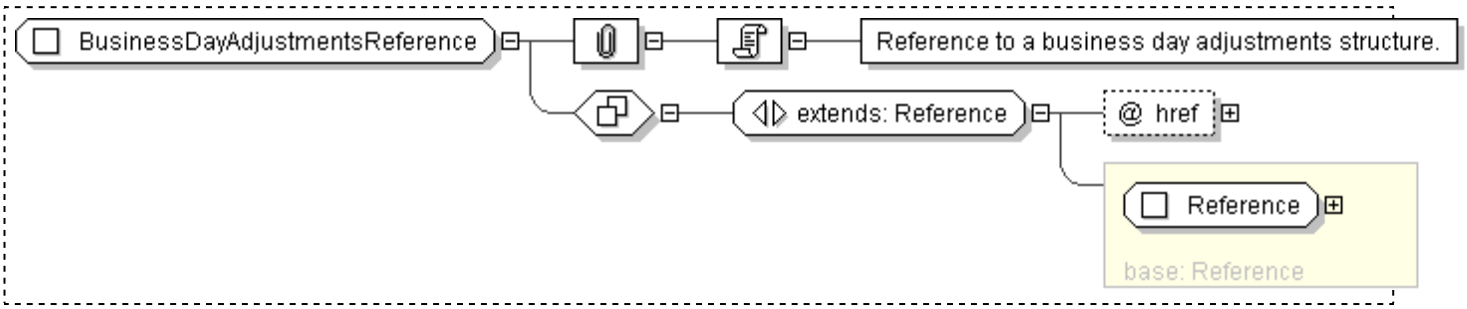
Super-types:	Reference < BusinessDayAdjustmentsReference (by extension)
Sub-types:	None

Name	BusinessDayAdjustmentsReference
Used by (from the same schema document)	Complex Type AdjustableDate2
Abstract	no
Documentation	Reference to a business day adjustments structure.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BusinessDayAdjustmentsReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="BusinessDayAdjustments"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationAgent

[Table of contents]

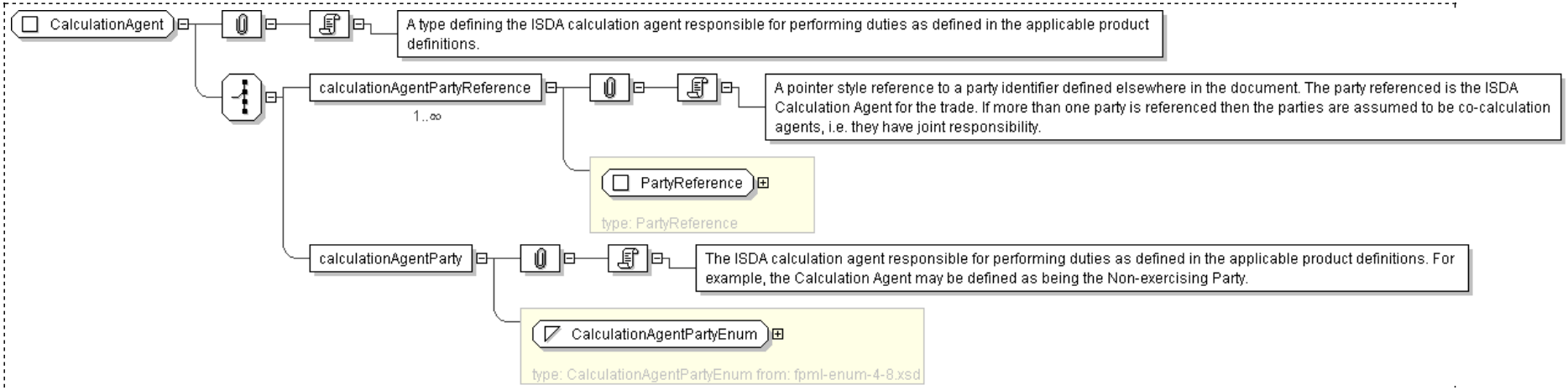
Super-types:	None
Sub-types:	None

Name	CalculationAgent
Abstract	no
Documentation	A type defining the ISDA calculation agent responsible for performing duties as defined in the applicable product definitions.

XML Instance Representation

```
<...>
  Start Choice [1]
    <calculationAgentPartyReference> PartyReference </calculationAgentPartyReference> [1..*]
    'A pointer style reference to a party identifier defined elsewhere in the document. The party referenced is the ISDA Calculation Agent for
    the trade. If more than one party is referenced then the parties are assumed to be co-calculation agents, i.e. they have joint
    responsibility.'

    <calculationAgentParty> CalculationAgentPartyEnum </calculationAgentParty> [1]
    'The ISDA calculation agent responsible for performing duties as defined in the applicable product definitions. For example, the Calculation
    Agent may be defined as being the Non-exercising Party.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationAgent">
  <xsd:choice>
    <xsd:element name="calculationAgentPartyReference" type=" PartyReference " maxOccurs="unbounded"/>
    <xsd:element name="calculationAgentParty" type=" CalculationAgentPartyEnum "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CalculationPeriodFrequency

[Table of contents]

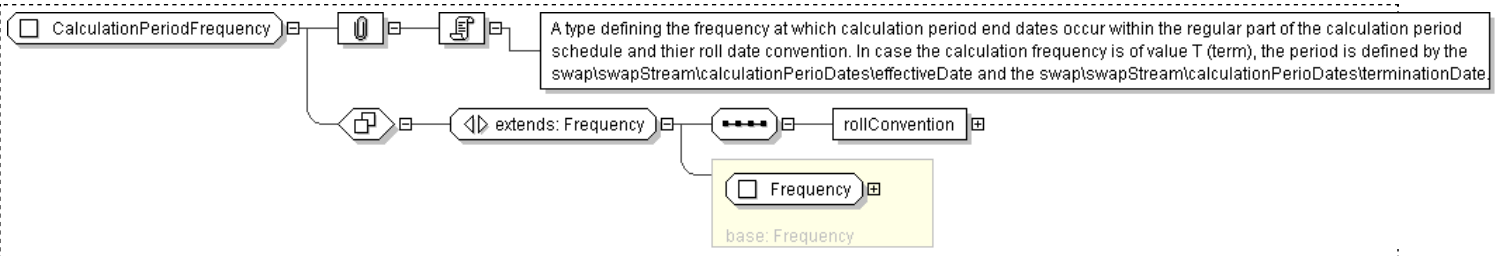
Super-types:	Frequency < CalculationPeriodFrequency (by extension)
Sub-types:	None

Name	CalculationPeriodFrequency
Used by (from the same schema document)	Complex Type PeriodicDates
Abstract	no
Documentation	A type defining the frequency at which calculation period end dates occur within the regular part of the calculation period schedule and thier roll date convention. In case the calculation frequency is of value T (term), the period is defined by the swap\swapStream\calculationPerioDates\effectiveDate and the swap\swapStream\calculationPerioDates\terminationDate.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to
   another date, e.g. -2 days. If the period value is T (Term) then periodMultiplier must contain the value 1.'

   <period> PeriodExtendedEnum </period> [1]
   'A time period, e.g. a day, week, month, year or term of the stream. If the periodMultiplier value is 0 (zero) then
   period must contain the value D (day).'

   <rollConvention> RollConventionEnum </rollConvention> [1]
   'Used in conjunction with a frequency and the regular period start date of a calculation period, determines each
   calculation period end date within the regular part of a calculation period schedule.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationPeriodFrequency">
  <xsd:complexContent>
    <xsd:extension base=" Frequency ">
      <xsd:sequence>
        <xsd:element name="rollConvention" type=" RollConventionEnum "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashflowType

[Table of contents]

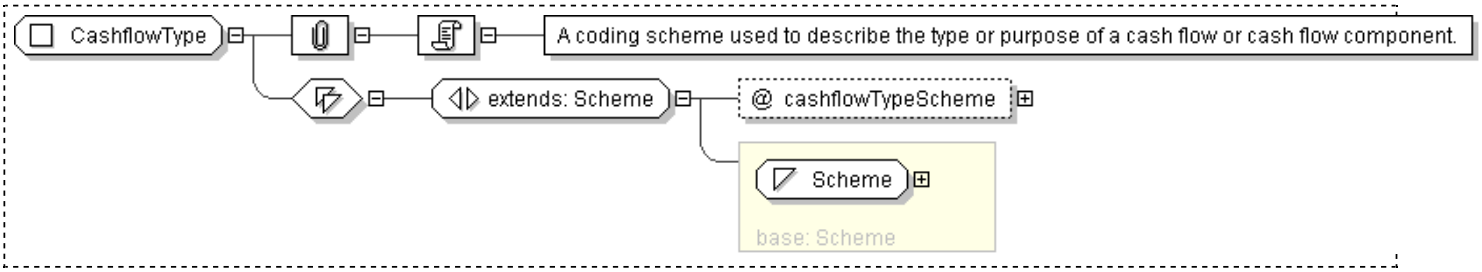
Super-types:	xsd:normalizedString < Scheme (by restriction) < CashflowType (by extension)
Sub-types:	None

Name	CashflowType
Abstract	no
Documentation	A coding scheme used to describe the type or purpose of a cash flow or cash flow component.

XML Instance Representation

```
<...
 cashflowTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashflowType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="cashflowTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/cashflow-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CashSettlementReferenceBanks

[Table of contents]

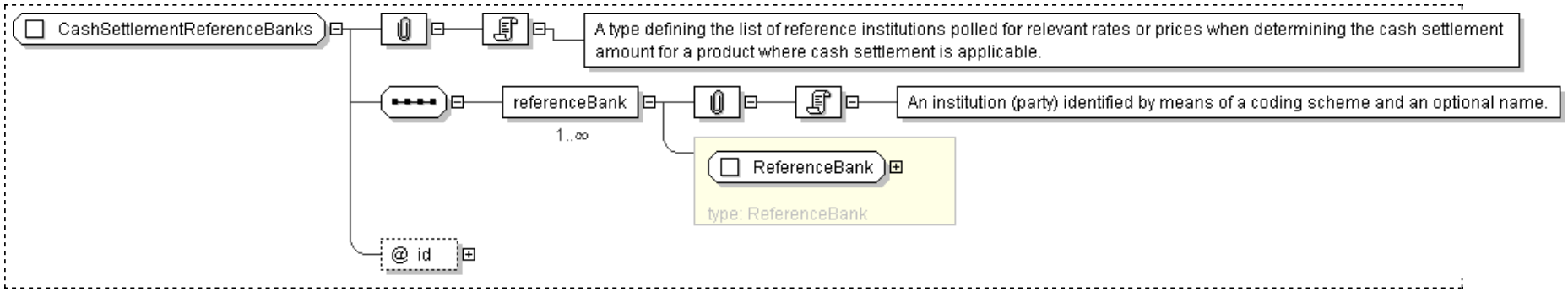
Super-types:	None
Sub-types:	None

Name	CashSettlementReferenceBanks
Used by (from the same schema document)	Complex Type SettlementRateSource
Abstract	no
Documentation	A type defining the list of reference institutions polled for relevant rates or prices when determining the cash settlement amount for a product where cash settlement is applicable.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <referenceBank> ReferenceBank </referenceBank> [1..*]
    'An institution (party) identified by means of a coding scheme and an optional name.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CashSettlementReferenceBanks">
  <xsd:sequence>
    <xsd:element name="referenceBank" type=" ReferenceBank " maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ClearanceSystem

[Table of contents]

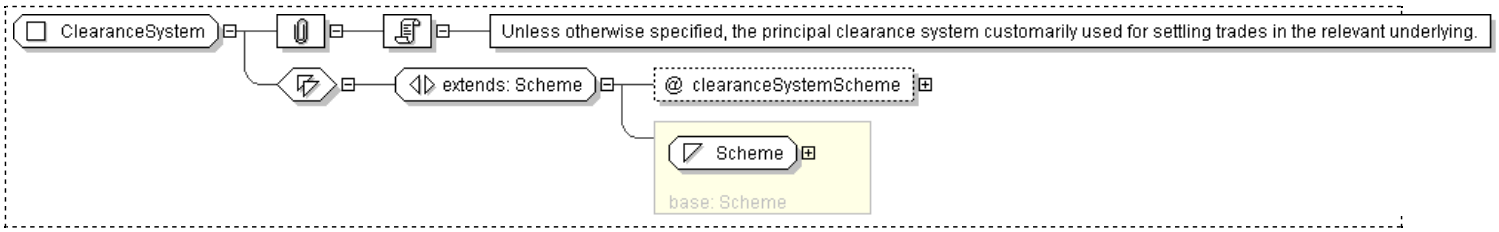
Super-types:	xsd:normalizedString < Scheme (by restriction) < ClearanceSystem (by extension)
Sub-types:	None

Name	ClearanceSystem
Abstract	no
Documentation	Unless otherwise specified, the principal clearance system customarily used for settling trades in the relevant underlying.

XML Instance Representation

```
<...
 clearanceSystemScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ClearanceSystem">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="clearanceSystemScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/clearance-system"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractualDefinitions

[Table of contents]

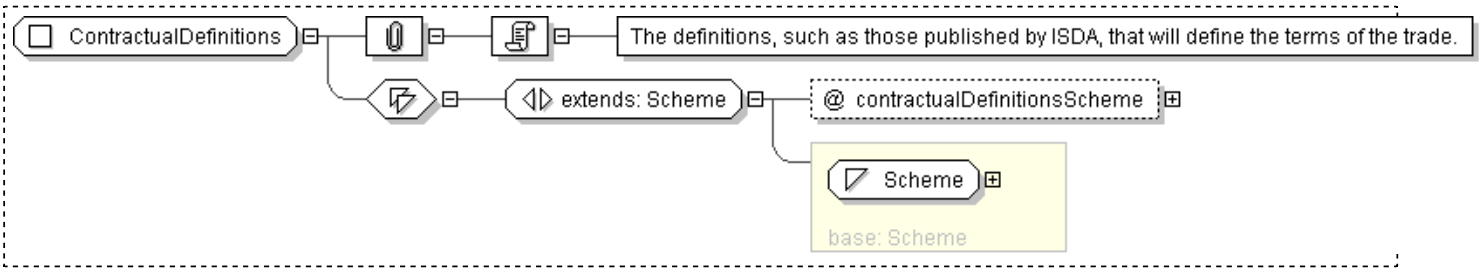
Super-types:	xsd:normalizedString < Scheme (by restriction) < ContractualDefinitions (by extension)
Sub-types:	None

Name	ContractualDefinitions
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	The definitions, such as those published by ISDA, that will define the terms of the trade.

XML Instance Representation

```
<...
 contractualDefinitionsScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractualDefinitions">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="contractualDefinitionsScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/contractual-definitions"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractualMatrix

[Table of contents]

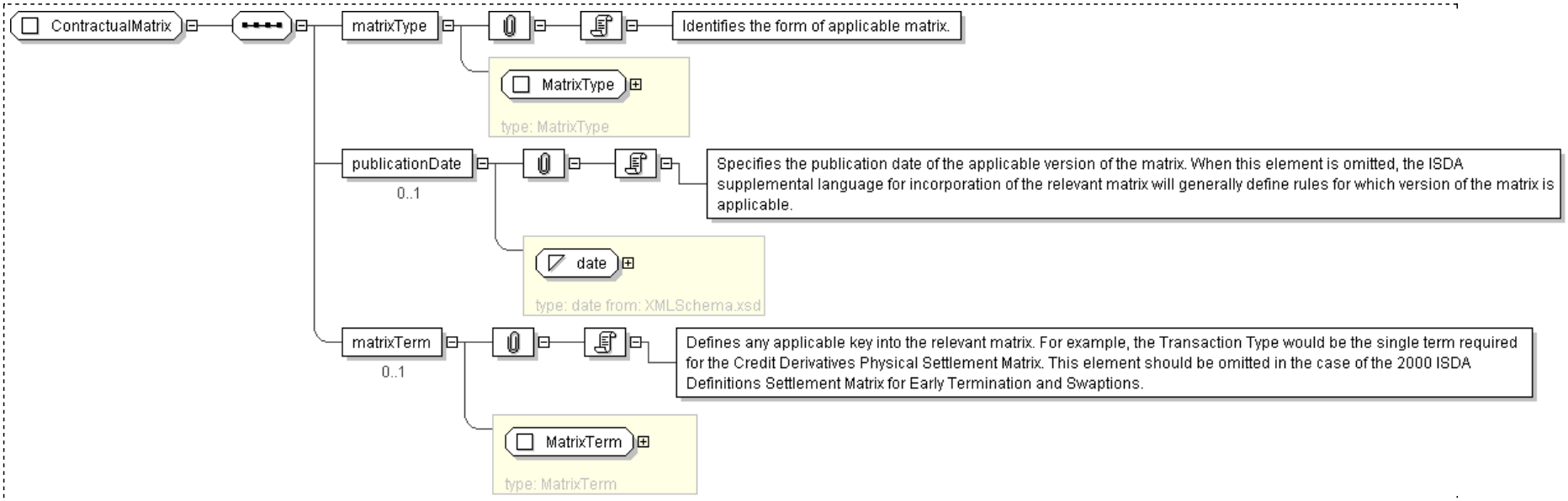
Super-types:	None
Sub-types:	None

Name	ContractualMatrix
Used by (from the same schema document)	Complex Type Documentation
Abstract	no

XML Instance Representation

```
<...>
   <matrixType> MatrixType </matrixType> [1]
   'Identifies the form of applicable matrix.'

   <publicationDate> xsd:date </publicationDate> [0..1]
   'Specifies the publication date of the applicable version of the matrix. When this element is omitted, the ISDA supplemental language
   for incorporation of the relevant matrix will generally define rules for which version of the matrix is applicable.'

   <matrixTerm> MatrixTerm </matrixTerm> [0..1]
   'Defines any applicable key into the relevant matrix. For example, the Transaction Type would be the single term required for the
   Credit Derivatives Physical Settlement Matrix. This element should be omitted in the case of the 2000 ISDA Definitions Settlement Matrix
   for Early Termination and Swaptions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractualMatrix">
   <xsd:sequence>
```

```
      <xsd:element name="matrixType" type=" MatrixType "/>
      <xsd:element name="publicationDate" type=" xsd:date " minOccurs="0"/>
      <xsd:element name="matrixTerm" type=" MatrixTerm " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractualSupplement

[Table of contents]

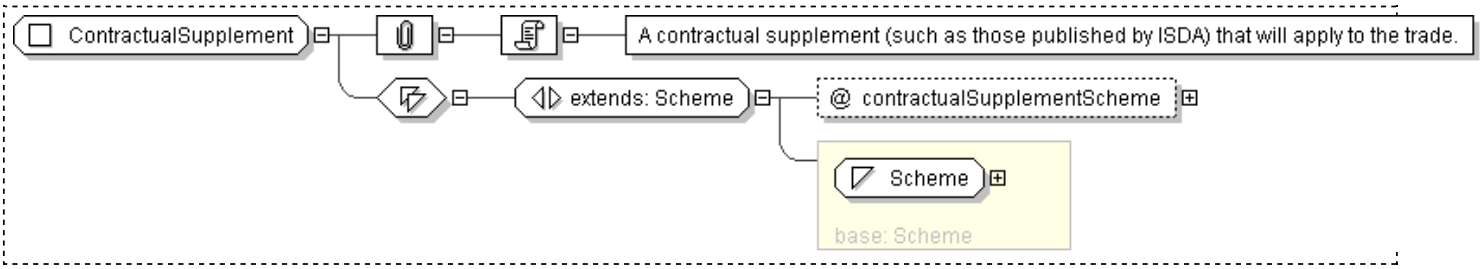
Super-types:	xsd:normalizedString < Scheme (by restriction) < ContractualSupplement (by extension)
Sub-types:	None

Name	ContractualSupplement
Used by (from the same schema document)	Complex Type ContractualTermsSupplement , Complex Type Documentation
Abstract	no
Documentation	A contractual supplement (such as those published by ISDA) that will apply to the trade.

XML Instance Representation

```
<...
 contractualSupplementScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractualSupplement">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="contractualSupplementScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/contractual-supplement"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ContractualTermsSupplement

[Table of contents]

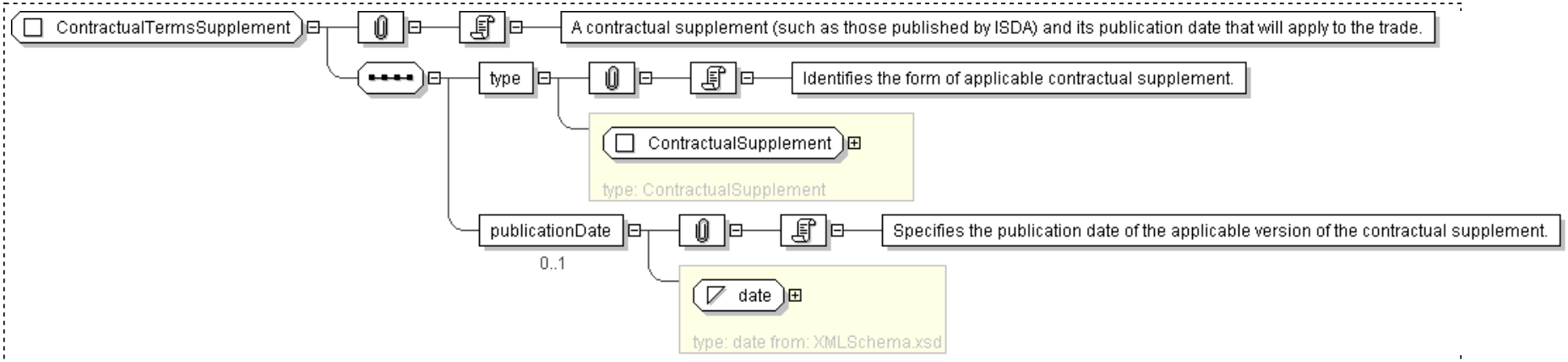
Super-types:	None
Sub-types:	None

Name	ContractualTermsSupplement
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	A contractual supplement (such as those published by ISDA) and its publication date that will apply to the trade.

XML Instance Representation

```
<...>
  <type> ContractualSupplement </type> [1]
  'Identifies the form of applicable contractual supplement.'

  <publicationDate> xsd:date </publicationDate> [0..1]
  'Specifies the publication date of the applicable version of the contractual supplement.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ContractualTermsSupplement">
  <xsd:sequence>
    <xsd:element name="type" type=" ContractualSupplement "/>
    <xsd:element name="publicationDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

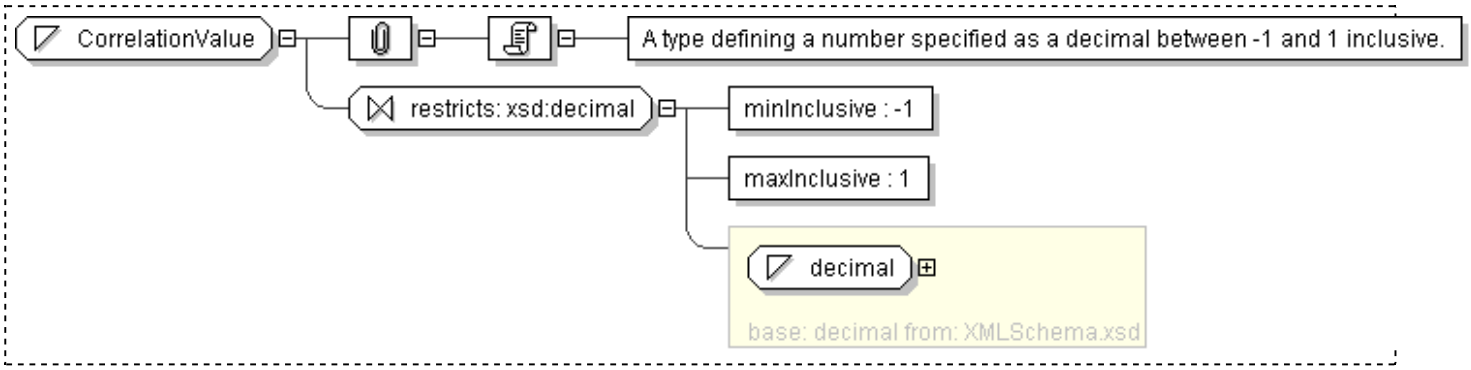
Simple Type: CorrelationValue

[Table of contents]

Super-types:	xsd:decimal < CorrelationValue (by restriction)
Sub-types:	None

Name	CorrelationValue
Content	• Base XSD Type: decimal • -1 <= *value* <= 1
Documentation	A type defining a number specified as a decimal between -1 and 1 inclusive.

Diagram



Schema Component Representation

```
<xsd:simpleType name="CorrelationValue">
  <xsd:restriction base=" xsd:decimal ">
    <xsd:minInclusive value="-1"/>
    <xsd:maxInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CorrespondentInformation

[Table of contents]

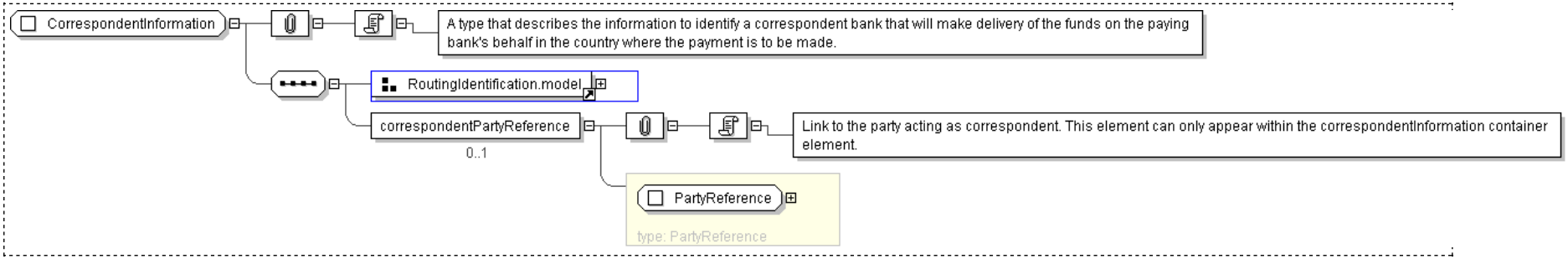
Super-types:	None
Sub-types:	None

Name	CorrespondentInformation
Used by (from the same schema document)	Complex Type SettlementInstruction
Abstract	no
Documentation	A type that describes the information to identify a correspondent bank that will make delivery of the funds on the paying bank's behalf in the country where the payment is to be made.

XML Instance Representation

```
<...>
   Start Choice [1]
      <routingIds> RoutingIds </routingIds> [1]
      'A set of unique identifiers for a party, eachone identifying the party within a payment system. The assumption is that each party will not have
      more than one identifier within the same payment system.'

      <routingExplicitDetails> RoutingExplicitDetails </routingExplicitDetails> [1]
      'A set of details that is used to identify a party involved in the routing of a payment when the party does not have a code that identifies it
      within one of the recognized payment systems.'

      <routingIdsAndExplicitDetails> RoutingIdsAndExplicitDetails </routingIdsAndExplicitDetails> [1]
      'A combination of coded payment system identifiers and details for physical addressing for a party involved in the routing of a payment.'

   End Choice
   <correspondentPartyReference> PartyReference </correspondentPartyReference> [0..1]
   'Link to the party acting as correspondent. This element can only appear within the correspondentInformation container element.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CorrespondentInformation">
   <xsd:sequence>
      <xsd:group ref=" RoutingIdentification.model "/>
      <xsd:element name="correspondentPartyReference" type=" PartyReference " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Country

[Table of contents]

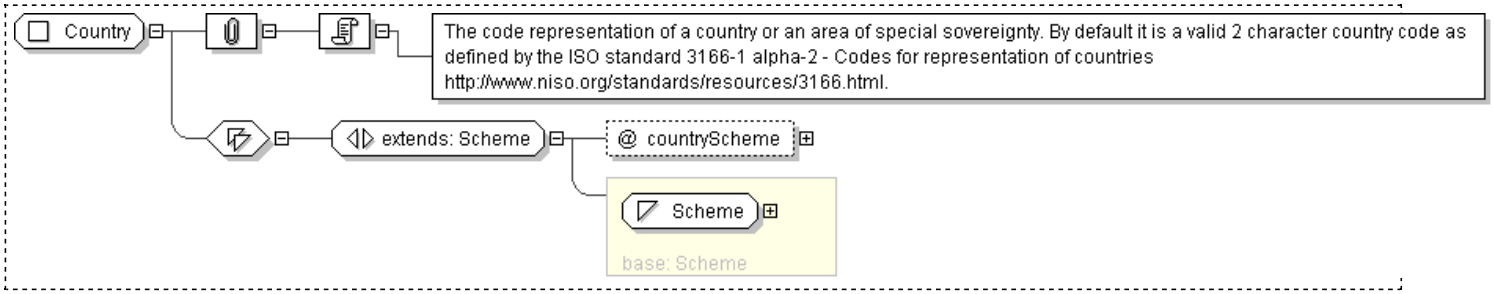
Super-types:	xsd:normalizedString < Scheme (by restriction) < Country (by extension)
Sub-types:	None

Name	Country
Used by (from the same schema document)	Complex Type Address
Abstract	no
Documentation	The code representation of a country or an area of special sovereignty. By default it is a valid 2 character country code as defined by the ISO standard 3166-1 alpha-2 - Codes for representation of countries http://www.niso.org/standards/resources/3166.html.

XML Instance Representation

```
<...
 countryScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Country">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="countryScheme" type=" xsd:anyURI " default="http://www.fpml.org/ext/iso3166"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditSeniority

[Table of contents]

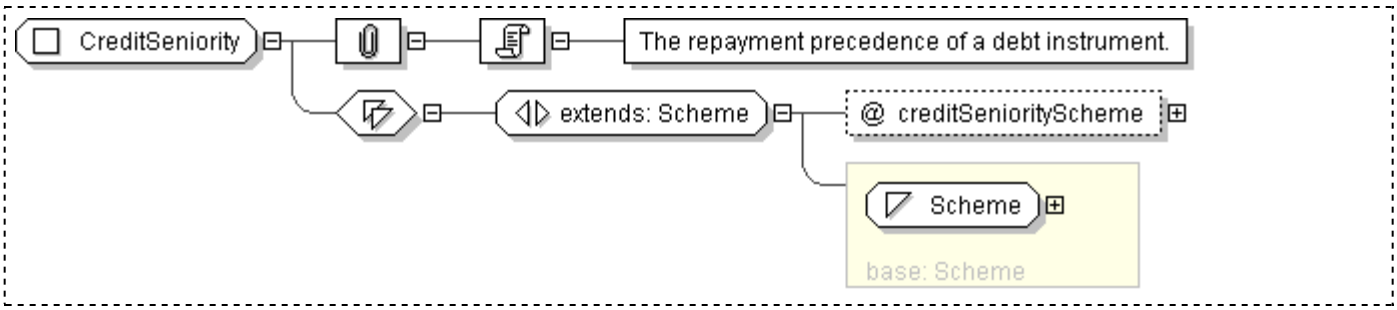
Super-types:	xsd:normalizedString < Scheme (by restriction) < CreditSeniority (by extension)
Sub-types:	None

Name	CreditSeniority
Abstract	no
Documentation	The repayment precedence of a debt instrument.

XML Instance Representation

```
<...
 creditSeniorityScheme=" xsd:anyURI [0..1]
'creditSeniorityTradingScheme overrides creditSeniorityScheme when the underlyer
defines the reference obligation used in a single name credit default swap trade.'

 ">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditSeniority">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="creditSeniorityScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/credit-seniority"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditSupportAgreement

[Table of contents]

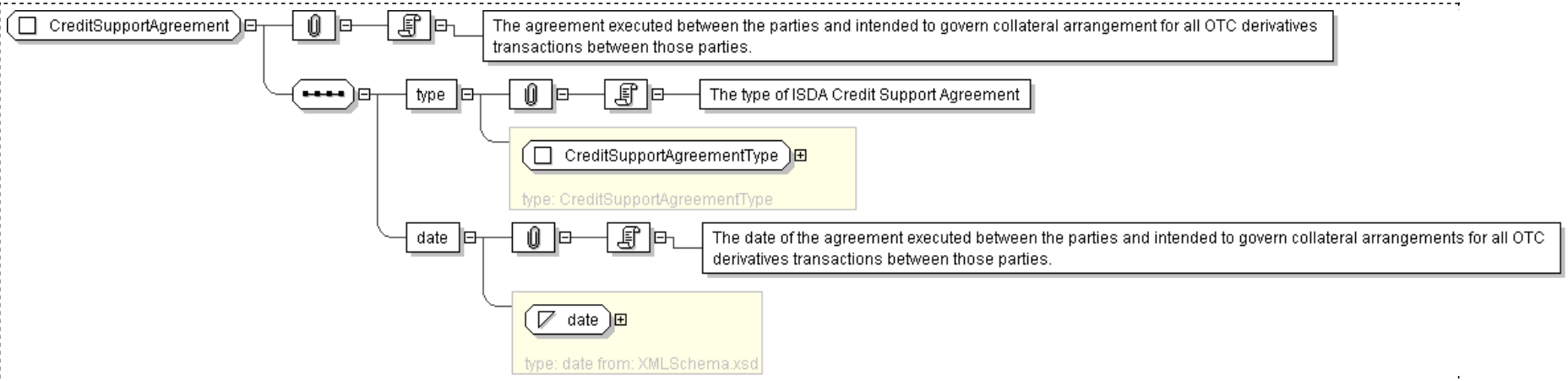
Super-types:	None
Sub-types:	None

Name	CreditSupportAgreement
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	The agreement executed between the parties and intended to govern collateral arrangement for all OTC derivatives transactions between those parties.

XML Instance Representation

```
<...>
   <type> CreditSupportAgreementType </type> [1]
   'The type of ISDA Credit Support Agreement'

   <date> xsd:date </date> [1]
   'The date of the agreement executed between the parties and intended to govern collateral arrangements for all OTC derivatives
   transactions between those parties.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditSupportAgreement">
   <xsd:sequence>
      <xsd:element name="type" type=" CreditSupportAgreementType "/>
      <xsd:element name="date" type=" xsd:date "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CreditSupportAgreementType

[Table of contents]

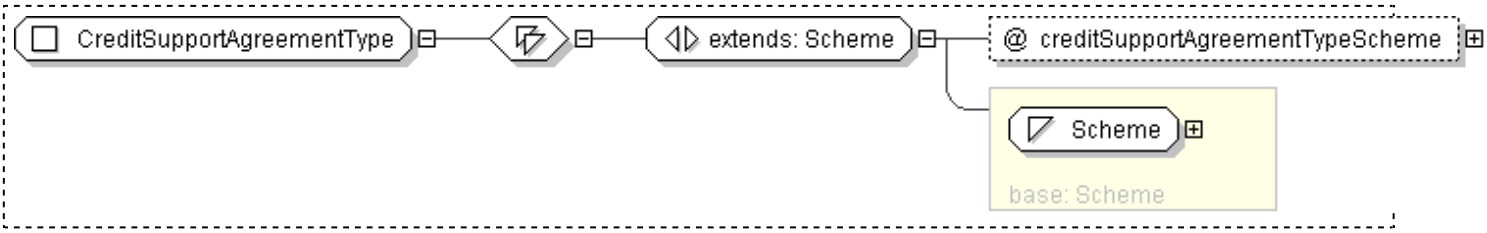
Super-types:	xsd:normalizedString < Scheme (by restriction) < CreditSupportAgreementType (by extension)
Sub-types:	None

Name	CreditSupportAgreementType
Used by (from the same schema document)	Complex Type CreditSupportAgreement
Abstract	no

XML Instance Representation

```
<...
 creditSupportAgreementTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CreditSupportAgreementType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="creditSupportAgreementTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/credit-support-agreement-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Currency

[Table of contents]

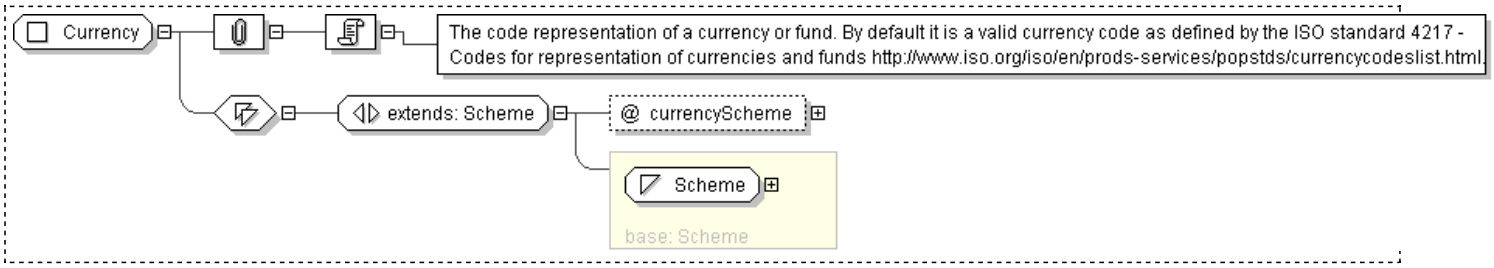
Super-types:	xsd:normalizedString < Scheme (by restriction) < Currency (by extension)
Sub-types:	None

Name	Currency
Used by (from the same schema document)	Complex Type AmountSchedule , Complex Type ExCashSettlement , Complex Type MoneyBase , Complex Type NonNegativeAmountSchedule , Complex Type PaymentCurrency , Complex Type PositiveAmountSchedule , Complex Type PricingStructure , Complex Type QuotedCurrencyPair , Complex Type QuotedCurrencyPair , Model Group SettlementAmountOrCurrency.model
Abstract	no
Documentation	The code representation of a currency or fund. By default it is a valid currency code as defined by the ISO standard 4217 - Codes for representation of currencies and funds http://www.iso.org/iso/en/prods-services/popstds/currencycodeslist.html.

XML Instance Representation

```
<...
 currencyScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Currency">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="currencyScheme" type=" xsd:anyURI " default="http://www.fpml.org/ext/iso4217-2001-08-15"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateList

[Table of contents]

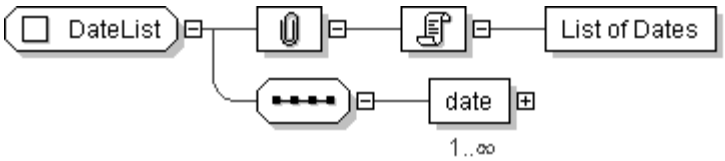
Super-types:	None
Sub-types:	None

Name	DateList
Abstract	no
Documentation	List of Dates

XML Instance Representation

```
<...>
   <date> xsd:date </date> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateList">
   <xsd:sequence>
      <xsd:element name="date" type=" xsd:date " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateOffset

[Table of contents]

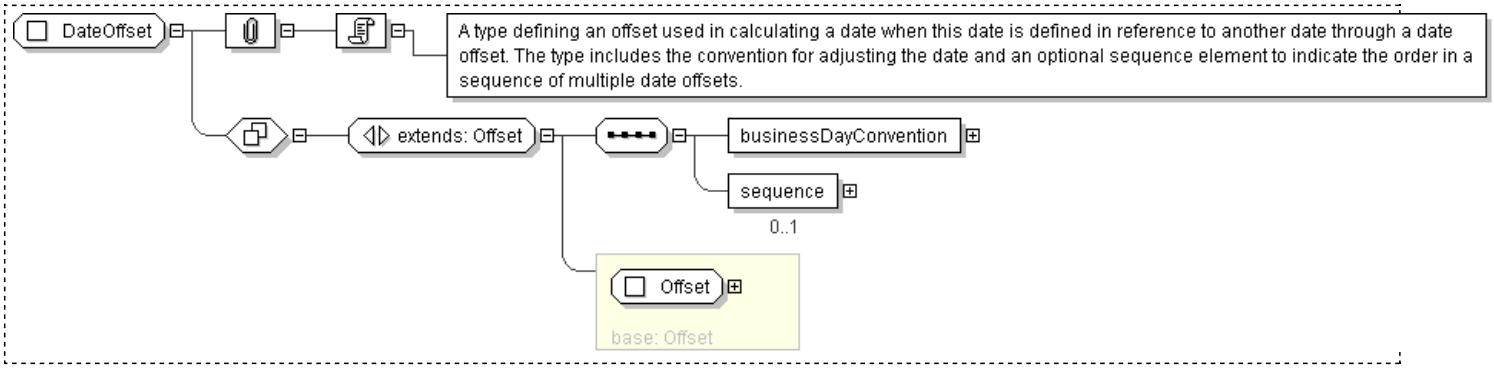
Super-types:	Period < Offset (by extension) < DateOffset (by extension)
Sub-types:	None

Name	DateOffset
Used by (from the same schema document)	Complex Type RelativeDateSequence
Abstract	no
Documentation	A type defining an offset used in calculating a date when this date is defined in reference to another date through a date offset. The type includes the convention for adjusting the date and an optional sequence element to indicate the order in a sequence of multiple date offsets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset
   relative to another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0 (zero)
   then period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
   'In the case of an offset specified as a number of days, this element defines whether consideration is
   given as to whether a day is a good business day or not. If a day type of business days is specified then
   non-business days are ignored when calculating the offset. The financial business centers to use for
   determination of business days are implied by the context in which this element is used. This element must
   only be included when the offset is specified as a number of days. If the offset is zero days then the
   dayType element should not be included.'

   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
   'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

   <sequence> xsd:positiveInteger </sequence> [0..1]
   'DEPRECATED Sequence in which the reference to the time period multiplier should be applied.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateOffset">
  <xsd:complexContent>
    <xsd:extension base=" Offset ">
      <xsd:sequence>
        <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
        <xsd:element name="sequence" type=" xsd:positiveInteger " minOccurs="0" deprecated="true"
        deprecatedReason="Ordering is already present in the XML instance document. It shouldn't be defined as
        an element as stated in the Architecture Specification"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateRange

[Table of contents]

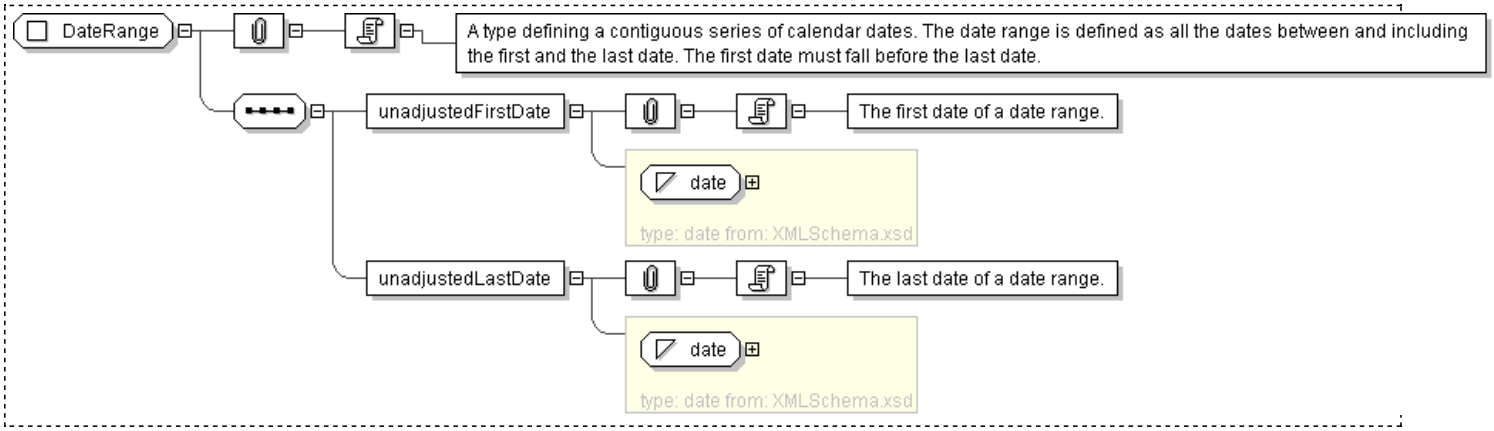
Super-types:	None
Sub-types:	• BusinessDateRange (by extension)

Name	DateRange
Used by (from the same schema document)	Complex Type RelativeDates
Abstract	no
Documentation	A type defining a contiguous series of calendar dates. The date range is defined as all the dates between and including the first and the last date. The first date must fall before the last date.

XML Instance Representation

```
<...>
  <unadjustedFirstDate> xsd:date </unadjustedFirstDate> [1]
  'The first date of a date range.'

  <unadjustedLastDate> xsd:date </unadjustedLastDate> [1]
  'The last date of a date range.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateRange">
  <xsd:sequence>
    <xsd:element name="unadjustedFirstDate" type=" xsd:date "/>
    <xsd:element name="unadjustedLastDate" type=" xsd:date "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateReference

[Table of contents]

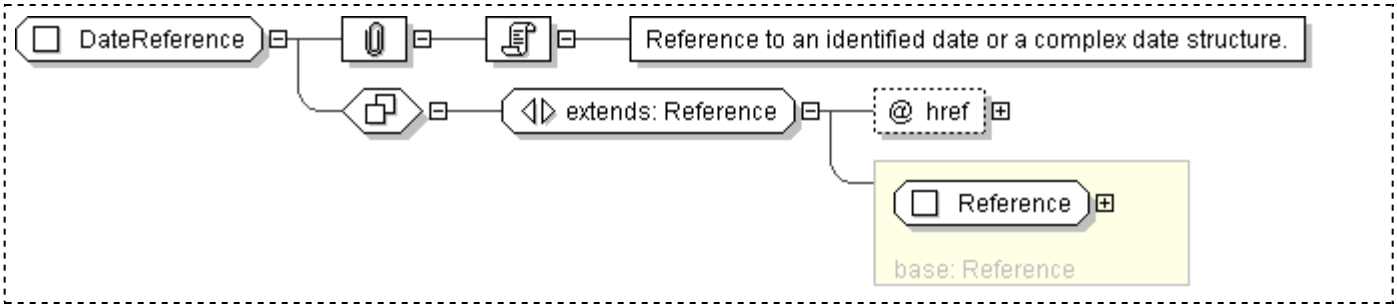
Super-types:	Reference < DateReference (by extension)
Sub-types:	None

Name	DateReference
Used by (from the same schema document)	Complex Type RelativeDateOffset , Complex Type RelativeDateSequence
Abstract	no
Documentation	Reference to an identified date or a complex date structure.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DateTimeList

[Table of contents]

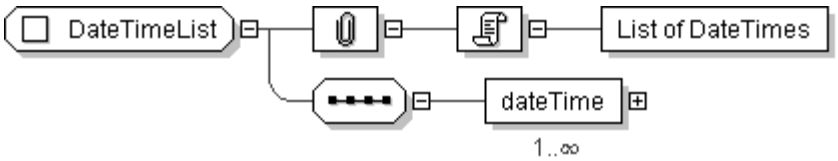
Super-types:	None
Sub-types:	None

Name	DateTimeList
Abstract	no
Documentation	List of DateTimes

XML Instance Representation

```
<...>
   <dateTime> xsd:dateTime </dateTime> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DateTimeList">
   <xsd:sequence>
      <xsd:element name="dateTime" type=" xsd:dateTime " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DayCountFraction

[Table of contents]

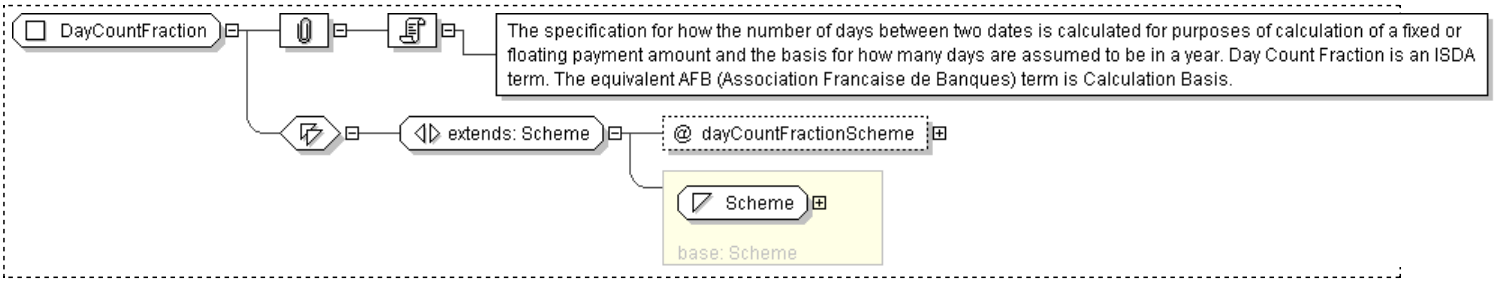
Super-types:	xsd:normalizedString < Scheme (by restriction) < DayCountFraction (by extension)
Sub-types:	None

Name	DayCountFraction
Abstract	no
Documentation	The specification for how the number of days between two dates is calculated for purposes of calculation of a fixed or floating payment amount and the basis for how many days are assumed to be in a year. Day Count Fraction is an ISDA term. The equivalent AFB (Association Francaise de Banques) term is Calculation Basis.

XML Instance Representation

```
<...
 dayCountFractionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DayCountFraction">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="dayCountFractionScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/day-count-fraction"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeterminationMethod

[Table of contents]

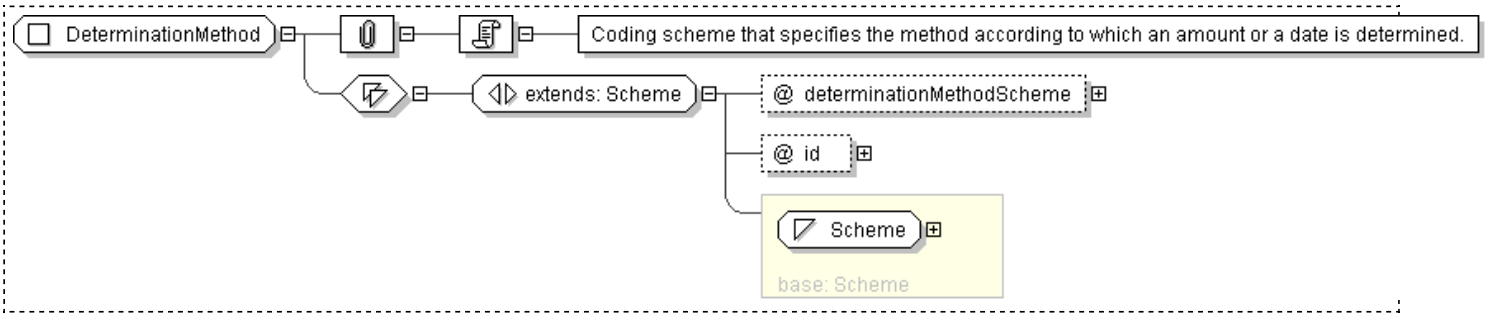
Super-types:	xsd:normalizedString < Scheme (by restriction) < DeterminationMethod (by extension)
Sub-types:	None

Name	DeterminationMethod
Used by (from the same schema document)	Complex Type PaymentCurrency , Complex Type SharedAmericanExercise
Abstract	no
Documentation	Coding scheme that specifies the method according to which an amount or a date is determined.

XML Instance Representation

```
<...
 determinationMethodScheme=" xsd:anyURI [0..1]"
 id=" xsd:ID [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeterminationMethod">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="determinationMethodScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/determination-method"/>
      <xsd:attribute name="id" type=" xsd:ID "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DeterminationMethodReference

[Table of contents]

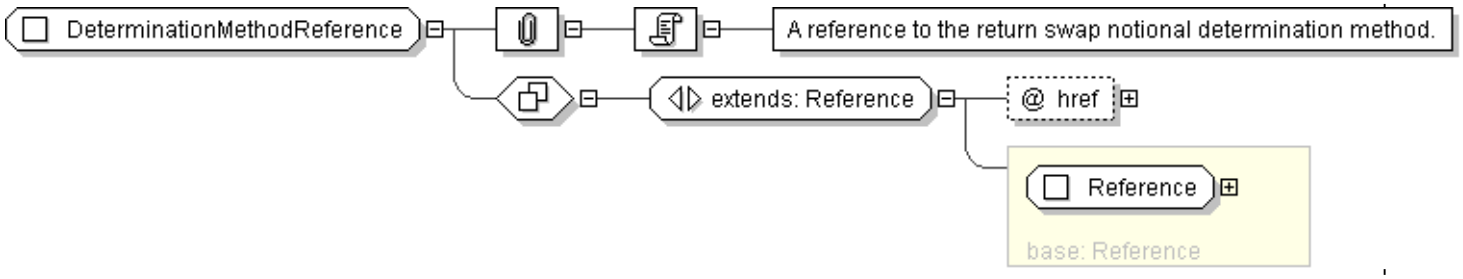
Super-types:	Reference < DeterminationMethodReference (by extension)
Sub-types:	None

Name	DeterminationMethodReference
Abstract	no
Documentation	A reference to the return swap notional determination method.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeterminationMethodReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="DeterminationMethod"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Documentation

[Table of contents]

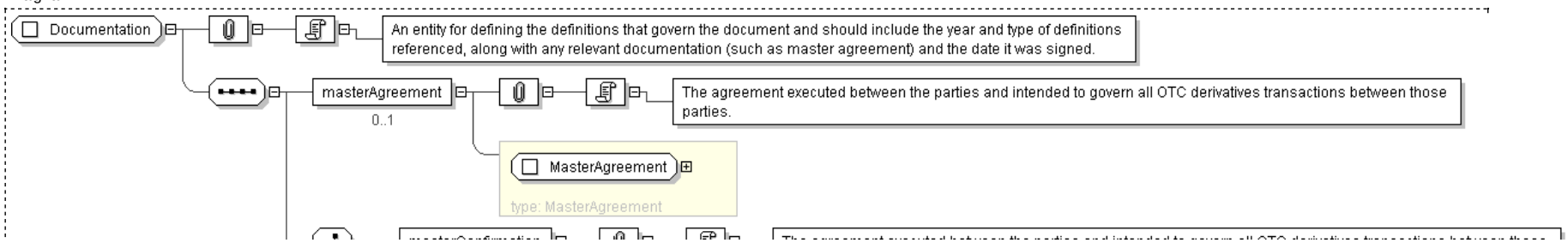
Super-types:	None
Sub-types:	None

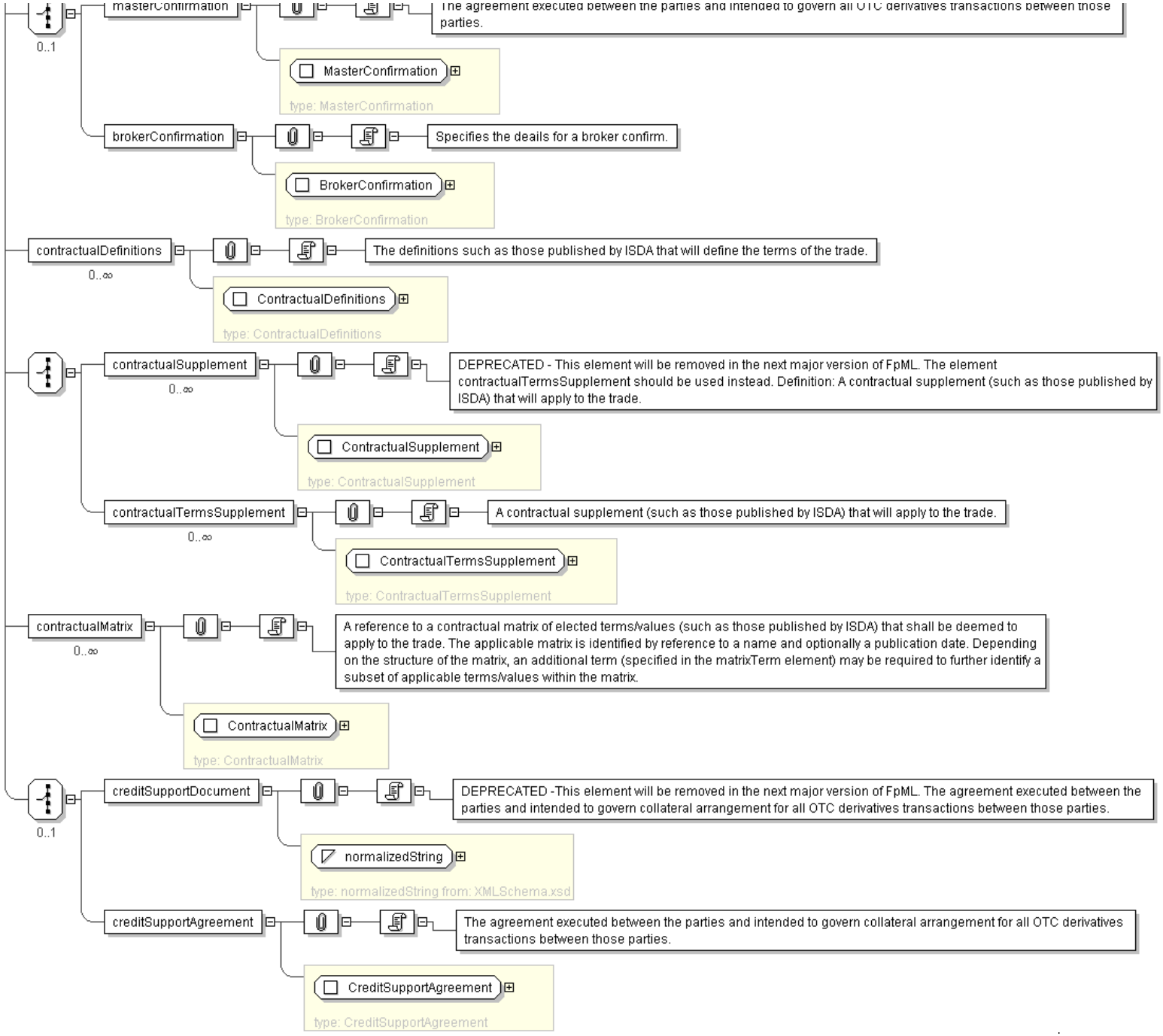
Name	Documentation
Abstract	no
Documentation	An entity for defining the definitions that govern the document and should include the year and type of definitions referenced, along with any relevant documentation (such as master agreement) and the date it was signed.

XML Instance Representation

```
<...>
  <masterAgreement> MasterAgreement </masterAgreement> [0..1]
   'The agreement executed between the parties and intended to govern all OTC derivatives transactions between those parties.'

  Start Choice [0..1]
    <masterConfirmation> MasterConfirmation </masterConfirmation> [1]
     'The agreement executed between the parties and intended to govern all OTC derivatives transactions between those parties.'

    <brokerConfirmation> BrokerConfirmation </brokerConfirmation> [1]
     'Specifies the deails for a broker confirm.'

  End Choice
  <contractualDefinitions> ContractualDefinitions </contractualDefinitions> [0..*]
   'The definitions such as those published by ISDA that will define the terms of the trade.'

  Start Choice [1]
    <contractualSupplement> ContractualSupplement </contractualSupplement> [0..*]
     'DEPRECATED - This element will be removed in the next major version of FpML. The element contractualTermsSupplement should be used instead.
     Definition: A contractual supplement (such as those published by ISDA) that will apply to the trade.'

    <contractualTermsSupplement> ContractualTermsSupplement </contractualTermsSupplement> [0..*]
     'A contractual supplement (such as those published by ISDA) that will apply to the trade.'

  End Choice
  <contractualMatrix> ContractualMatrix </contractualMatrix> [0..*]
   'A reference to a contractual matrix of elected terms/values (such as those published by ISDA) that shall be deemed to apply to the trade. The
   applicable matrix is identified by reference to a name and optionally a publication date. Depending on the structure of the matrix, an additional
   term (specified in the matrixTerm element) may be required to further identify a subset of applicable terms/values within the matrix.'

  Start Choice [0..1]
    <creditSupportDocument> xsd:normalizedString </creditSupportDocument> [1]
     'DEPRECATED -This element will be removed in the next major version of FpML. The agreement executed between the parties and intended to govern
     collateral arrangement for all OTC derivatives transactions between those parties.'

    <creditSupportAgreement> CreditSupportAgreement </creditSupportAgreement> [1]
     'The agreement executed between the parties and intended to govern collateral arrangement for all OTC derivatives transactions between those
     parties.'

  End Choice
</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="Documentation">
  <xsd:sequence>
    <xsd:element name="masterAgreement" type=" MasterAgreement " minOccurs="0"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="masterConfirmation" type=" MasterConfirmation "/>
```

```
        <xsd:element name="brokerConfirmation" type=" BrokerConfirmation "/>
      </xsd:choice>
      <xsd:element name="contractualDefinitions" type=" ContractualDefinitions " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:choice>
        <xsd:element name="contractualSupplement" type=" ContractualSupplement " minOccurs="0" maxOccurs="unbounded" deprecated="true"
        deprecatedReason="The contractualTermsSupplement includes the publication date, which was not present in the contractualSupplement"/>
        <xsd:element name="contractualTermsSupplement" type=" ContractualTermsSupplement " minOccurs="0" maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:element name="contractualMatrix" type=" ContractualMatrix " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:choice minOccurs="0">
        <xsd:element name="creditSupportDocument" type=" xsd:normalizedString " deprecated="true" deprecatedReason="Created a new creditSupportAgreement
        element of type CreditSupportAgreement to replace the creditSupportDocument element."/>
        <xsd:element name="creditSupportAgreement" type=" CreditSupportAgreement "/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Empty

[Table of contents]

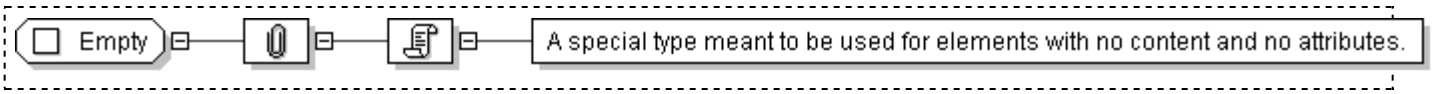
Super-types:	None
Sub-types:	None

Name	Empty
Abstract	no
Documentation	A special type meant to be used for elements with no content and no attributes.

XML Instance Representation

```
<.../>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Empty"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EntityId

[Table of contents]

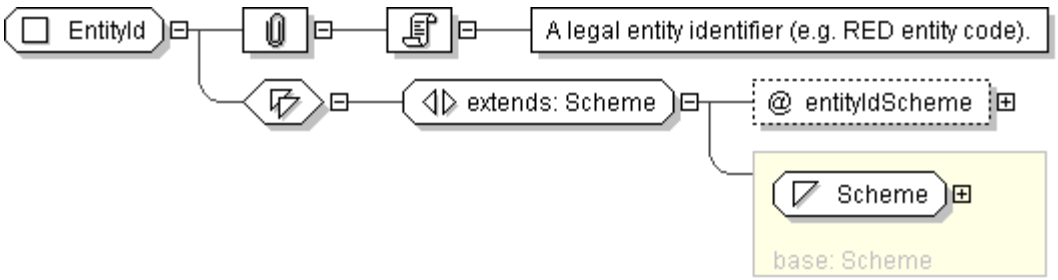
Super-types:	xsd:normalizedString < Scheme (by restriction) < EntityId (by extension)
Sub-types:	None

Name	EntityId
Used by (from the same schema document)	Complex Type LegalEntity , Complex Type LegalEntity
Abstract	no
Documentation	A legal entity identifier (e.g. RED entity code).

XML Instance Representation

```
<...
 entityIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EntityId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="entityIdScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/spec/2003/entity-id-RED-1-0"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EntityName

[Table of contents]

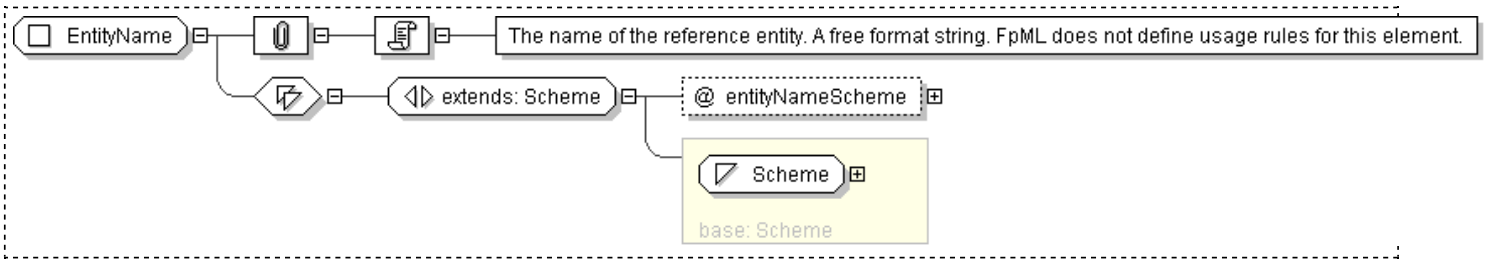
Super-types:	xsd:normalizedString < Scheme (by restriction) < EntityName (by extension)
Sub-types:	None

Name	EntityName
Used by (from the same schema document)	Complex Type LegalEntity
Abstract	no
Documentation	The name of the reference entity. A free format string. FpML does not define usage rules for this element.

XML Instance Representation

```
<...
 entityNameScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EntityName">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="entityNameScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/spec/2003/entity-name-RED-1-0"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: EuropeanExercise

[Table of contents]

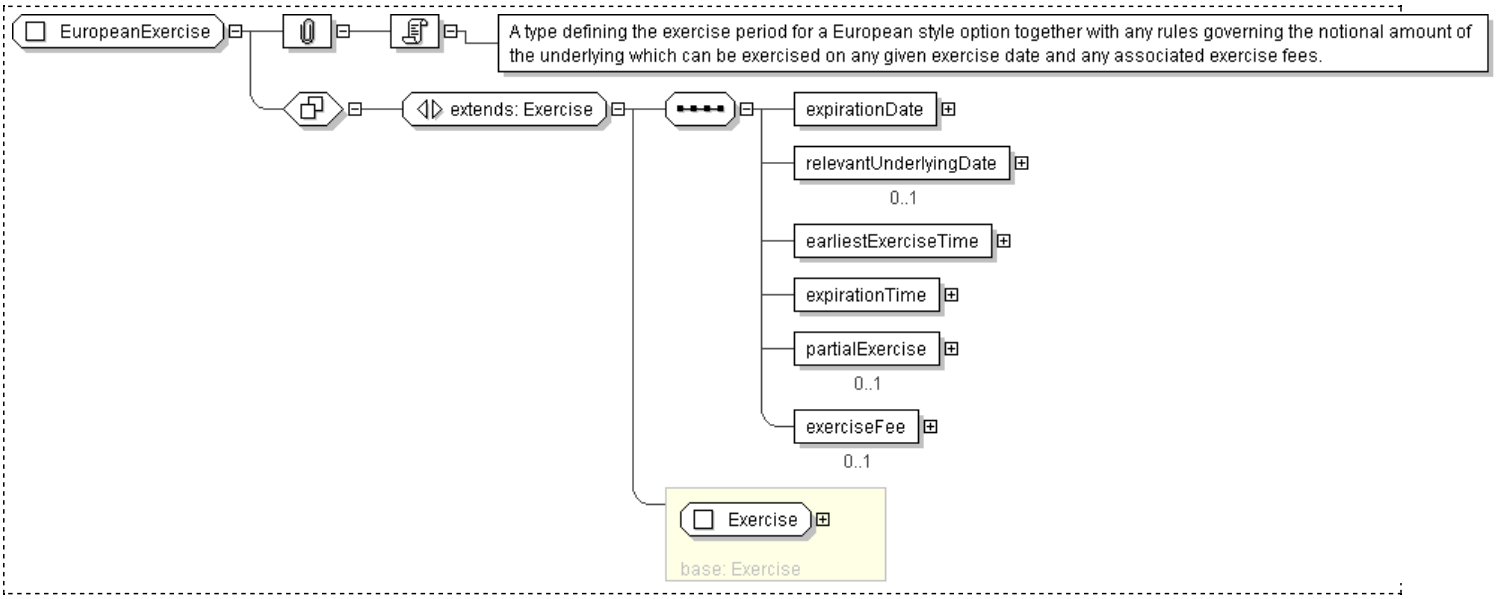
Super-types:	Exercise < EuropeanExercise (by extension)
Sub-types:	None

Name	EuropeanExercise
Used by (from the same schema document)	Element europeanExercise
Abstract	no
Documentation	A type defining the exercise period for a European style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
   'The last day within an exercise period for an American style option. For a European style option it is the
   only day within the exercise period.'

   <relevantUnderlyingDate> AdjustableOrRelativeDates </relevantUnderlyingDate> [0..1]
   'The daye on the underlying set by the exercise of an option. What this date is depends on the option (e.g.
   in a swaption it is the effective date, in an extendible/cancelable provision it is the termination date).'

   <earliestExerciseTime> BusinessCenterTime </earliestExerciseTime> [1]
   'The earliest time at which notice of exercise can be given by the buyer to the seller (or seller\'s agent)
   i) on the expriation date, in the case of a European style option, (ii) on each bermuda option exercise date
   and the expiration date, in the case of a Bermuda style option the commencement date to, and including, the
   expiration date , in the case of an American option.'

   <expirationTime> BusinessCenterTime </expirationTime> [1]
   'The latest time for exercise on expirationDate.'

   <partialExercise> PartialExercise </partialExercise> [0..1]
   'As defined in the 2000 ISDA Definitions, Section 12.3. Partial Exercise, the buyer of the option has the
   right to exercise all or less than all the notional amount of the underlying swap on the expiration date, but
   may not exercise less than the minimum notional amount, and if an integral multiple amount is specified, the
   notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.'

   <exerciseFee> ExerciseFee </exerciseFee> [0..1]
   'A fee to be paid on exercise. This could be represented as an amount or a rate and notional reference on
   which to apply the rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EuropeanExercise">
  <xsd:complexContent>
```

```
      <xsd:extension base=" Exercise ">
        <xsd:sequence>
          <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
          <xsd:element name="relevantUnderlyingDate" type=" AdjustableOrRelativeDates " minOccurs="0"/>
          <xsd:element name="earliestExerciseTime" type=" BusinessCenterTime "/>
          <xsd:element name="expirationTime" type=" BusinessCenterTime "/>
          <xsd:element name="partialExercise" type=" PartialExercise " minOccurs="0"/>
          <xsd:element name="exerciseFee" type=" ExerciseFee " minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExchangeId

[Table of contents]

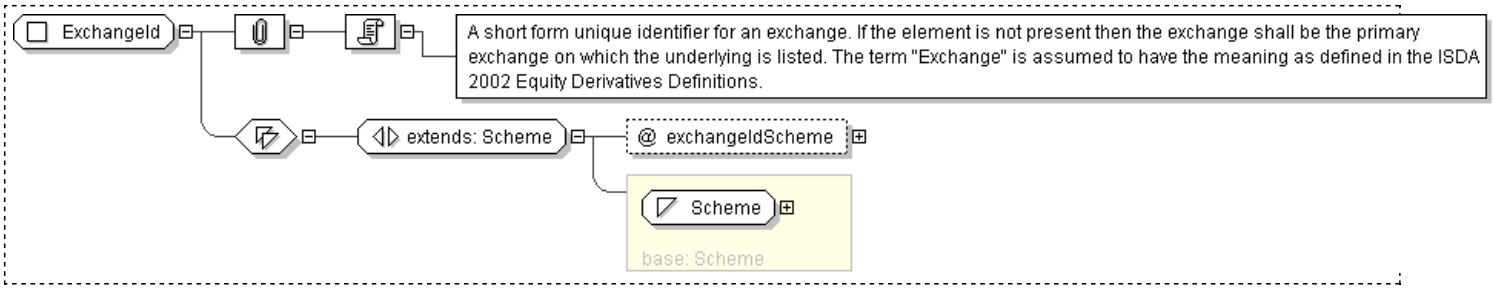
Super-types:	xsd:normalizedString < Scheme (by restriction) < ExchangeId (by extension)
Sub-types:	None

Name	ExchangeId
Abstract	no
Documentation	A short form unique identifier for an exchange. If the element is not present then the exchange shall be the primary exchange on which the underlying is listed. The term "Exchange" is assumed to have the meaning as defined in the ISDA 2002 Equity Derivatives Definitions.

XML Instance Representation

```
<...
 exchangeIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExchangeId">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="exchangeIdScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/spec/2002/exchange-id-MIC-1-0"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Exercise

[Table of contents]

Super-types: None

Sub-types:

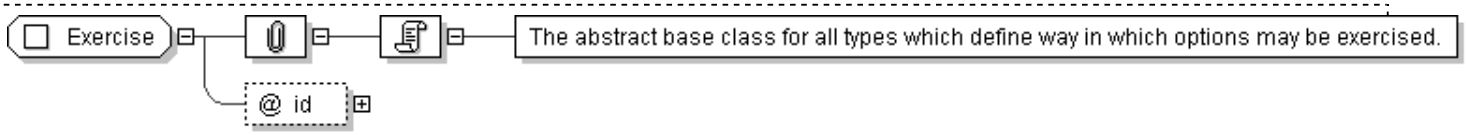
- AmericanExercise (by extension)
- BermudaExercise (by extension)
- EuropeanExercise (by extension)
- SharedAmericanExercise (by extension)

Name	Exercise
Used by (from the same schema document)	Element exercise
Abstract	no
Documentation	The abstract base class for all types which define way in which options may be exercised.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Exercise">
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExerciseFee

[Table of contents]

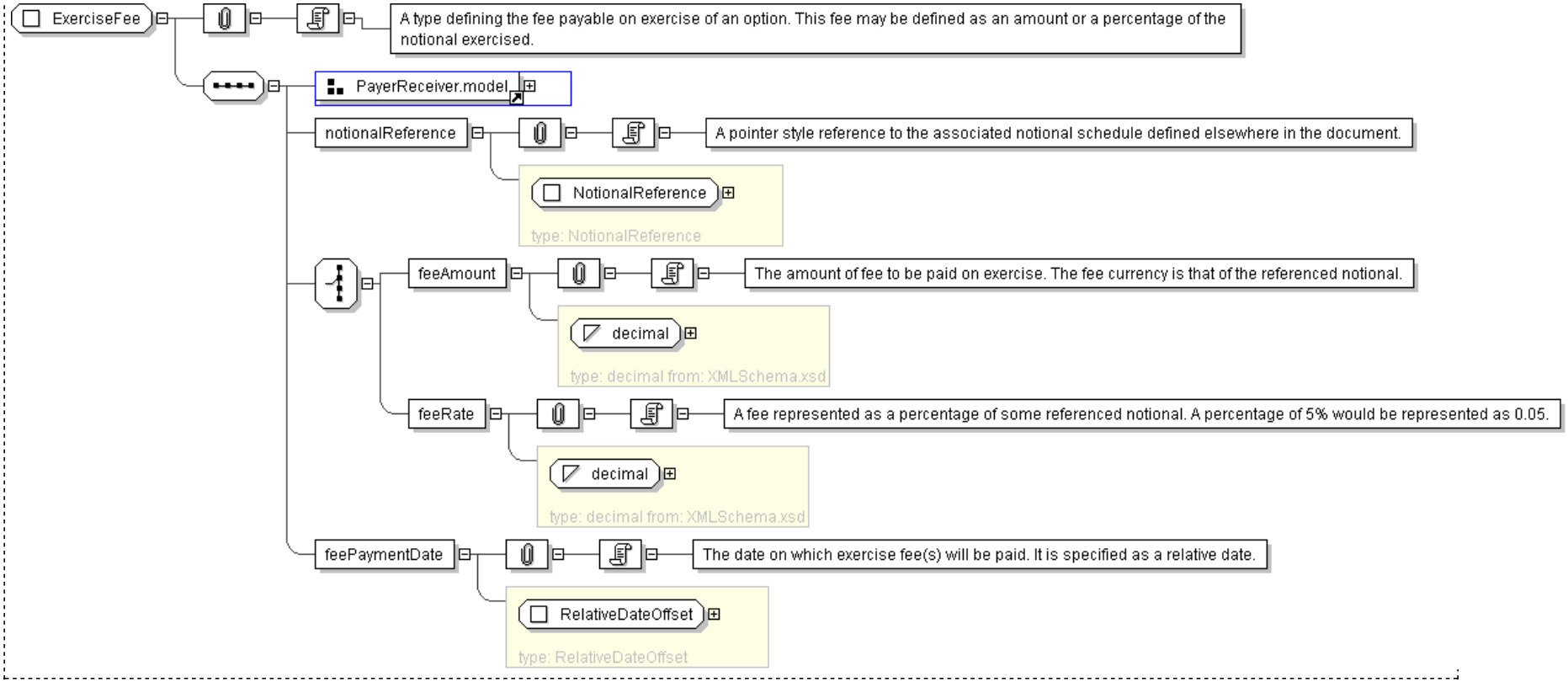
Super-types:	None
Sub-types:	None

Name	ExerciseFee
Used by (from the same schema document)	Complex Type EuropeanExercise
Abstract	no
Documentation	A type defining the fee payable on exercise of an option. This fee may be defined as an amount or a percentage of the notional exercised.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <notionalReference> NotionalReference </notionalReference> [1]
   'A pointer style reference to the associated notional schedule defined elsewhere in the document.'

   Start Choice [1]
      <feeAmount> xsd:decimal </feeAmount> [1]
      'The amount of fee to be paid on exercise. The fee currency is that of the referenced notional.'

      <feeRate> xsd:decimal </feeRate> [1]
      'A fee represented as a percentage of some referenced notional. A percentage of 5% would be represented as 0.05.'

   End Choice
   <feePaymentDate> RelativeDateOffset </feePaymentDate> [1]
   'The date on which exercise fee(s) will be paid. It is specified as a relative date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExerciseFee">
  <xsd:sequence>
    <xsd:group ref=" PayerReceiver.model "/>
    <xsd:element name="notionalReference" type=" NotionalReference "/>
    <xsd:choice>
      <xsd:element name="feeAmount" type=" xsd:decimal "/>
      <xsd:element name="feeRate" type=" xsd:decimal "/>
    </xsd:choice>
    <xsd:element name="feePaymentDate" type=" RelativeDateOffset "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExerciseFeeSchedule

[Table of contents]

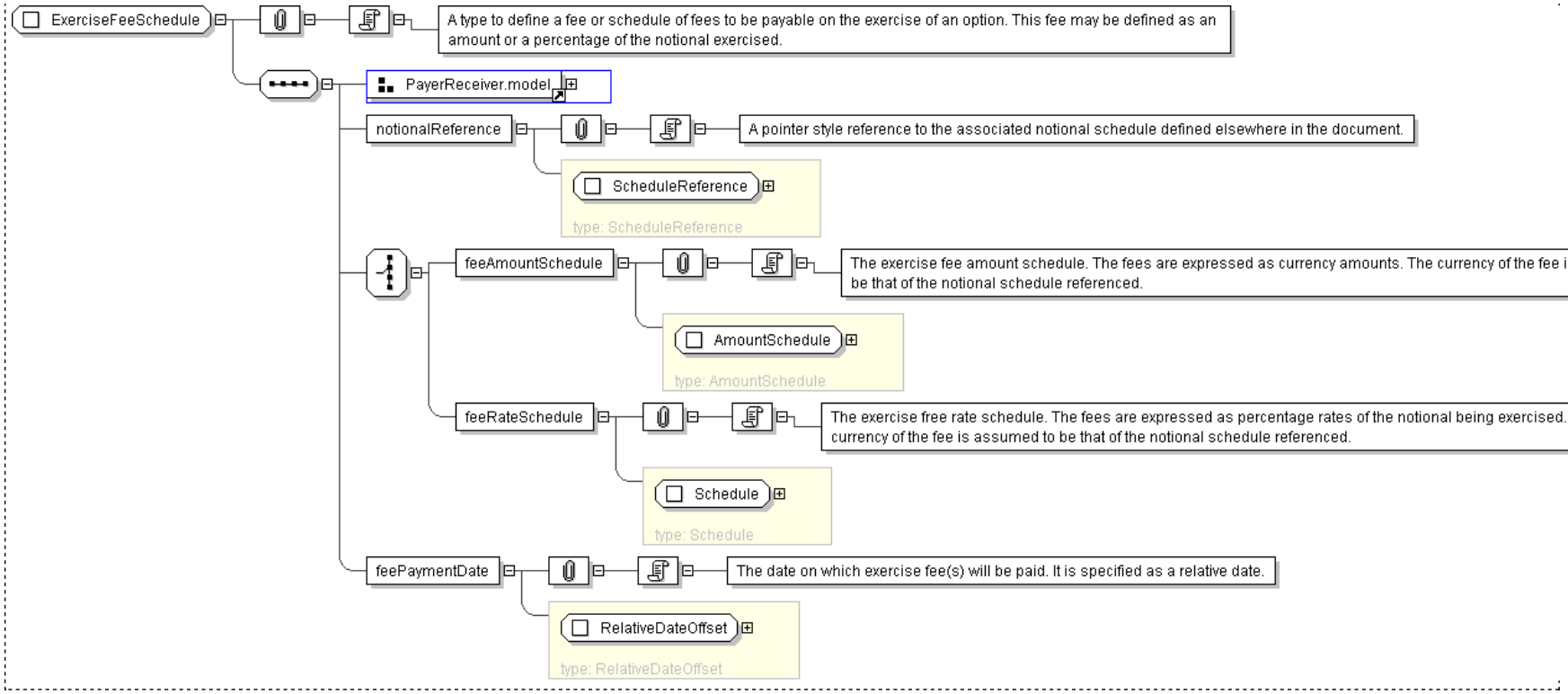
Super-types:	None
Sub-types:	None

Name	ExerciseFeeSchedule
Used by (from the same schema document)	Complex Type AmericanExercise , Complex Type BermudaExercise
Abstract	no
Documentation	A type to define a fee or schedule of fees to be payable on the exercise of an option. This fee may be defined as an amount or a percentage of the notional exercised.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this structure.'

   <notionalReference> ScheduleReference </notionalReference> [1]
    'A pointer style reference to the associated notional schedule defined elsewhere in the document.'

   Start Choice [1]
      <feeAmountSchedule> AmountSchedule </feeAmountSchedule> [1]
       'The exercise fee amount schedule. The fees are expressed as currency amounts. The currency of the fee is assumed to be that of the notional
       schedule referenced.'

      <feeRateSchedule> Schedule </feeRateSchedule> [1]
       'The exercise free rate schedule. The fees are expressed as percentage rates of the notional being exercised. The currency of the fee is assumed to
       be that of the notional schedule referenced.'

   End Choice
   <feePaymentDate> RelativeDateOffset </feePaymentDate> [1]
    'The date on which exercise fee(s) will be paid. It is specified as a relative date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExerciseFeeSchedule">
  <xsd:sequence>
    <xsd:group ref=" PayerReceiver.model "/>
    <xsd:element name="notionalReference" type=" ScheduleReference "/>
    <xsd:choice>
      <xsd:element name="feeAmountSchedule" type=" AmountSchedule "/>
      <xsd:element name="feeRateSchedule" type=" Schedule "/>
    </xsd:choice>
    <xsd:element name="feePaymentDate" type=" RelativeDateOffset "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExerciseNotice

[Table of contents]

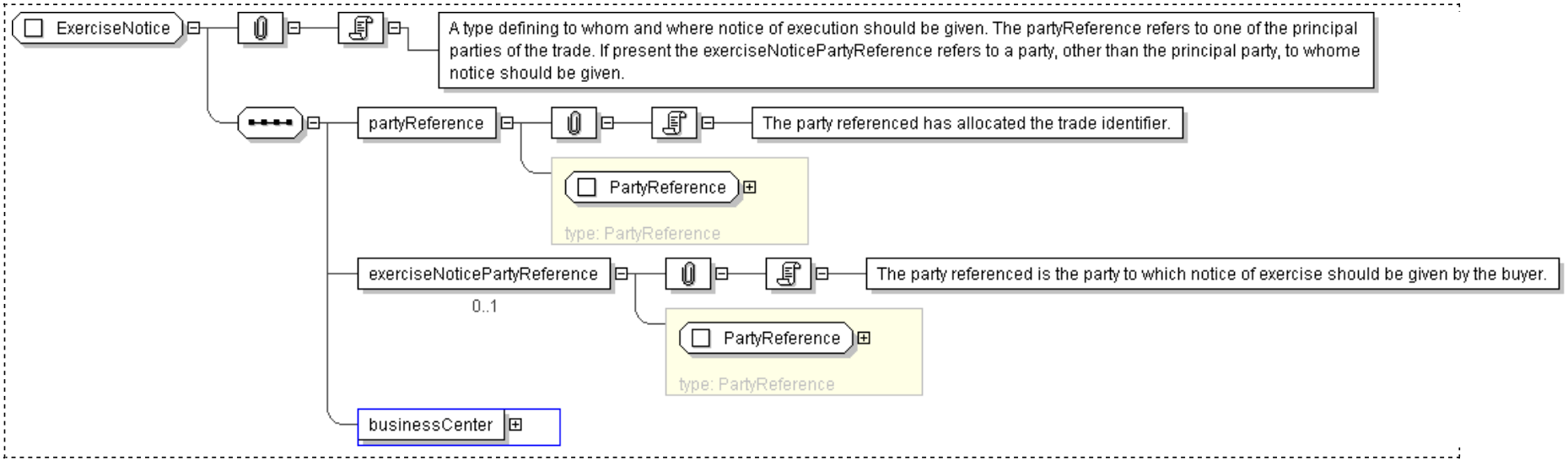
Super-types:	None
Sub-types:	None

Name	ExerciseNotice
Used by (from the same schema document)	Complex Type ManualExercise
Abstract	no
Documentation	A type defining to whom and where notice of execution should be given. The partyReference refers to one of the principal parties of the trade. If present the exerciseNoticePartyReference refers to a party, other than the principal party, to whome notice should be given.

XML Instance Representation

```
<...>
   <partyReference> PartyReference </partyReference> [1]
   'The party referenced has allocated the trade identifier.'

   <exerciseNoticePartyReference> PartyReference </exerciseNoticePartyReference> [0..1]
   'The party referenced is the party to which notice of exercise should be given by the buyer.'

   <businessCenter> BusinessCenter </businessCenter> [1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExerciseNotice">
```

```
  <xsd:sequence>
    <xsd:element name="partyReference" type=" PartyReference "/>
    <xsd:element name="exerciseNoticePartyReference" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="businessCenter" type=" BusinessCenter "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ExerciseProcedure

[Table of contents]

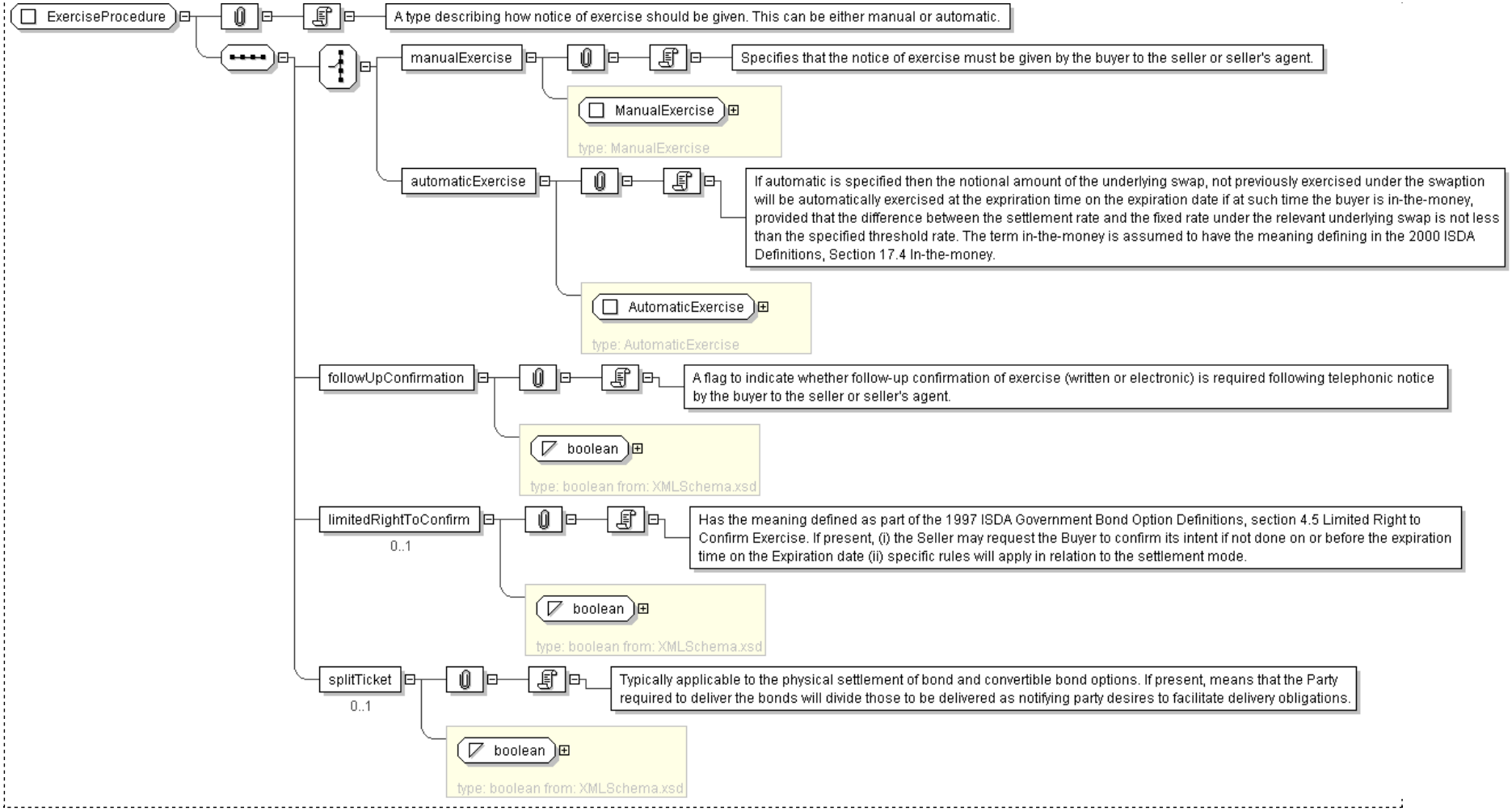
Super-types:	None
Sub-types:	None

Name	ExerciseProcedure
Abstract	no
Documentation	A type describing how notice of exercise should be given. This can be either manual or automatic.

XML Instance Representation

```
<...>
   Start Choice [1]
      <manualExercise> ManualExercise </manualExercise> [1]
      'Specifies that the notice of exercise must be given by the buyer to the seller or seller\'s agent.'

      <automaticExercise> AutomaticExercise </automaticExercise> [1]
      'If automatic is specified then the notional amount of the underlying swap, not previously exercised under the swaption will be automatically
      exercised at the expriration time on the expiration date if at such time the buyer is in-the-money, provided that the difference between the
      settlement rate and the fixed rate under the relevant underlying swap is not less than the specified threshold rate. The term in-the-money is
      assumed to have the meaning defining in the 2000 ISDA Definitions, Section 17.4 In-the-money.'

   End Choice
   <followUpConfirmation> xsd:boolean </followUpConfirmation> [1]
   'A flag to indicate whether follow-up confirmation of exercise (written or electronic) is required following telephonic notice by the buyer to the
   seller or seller\'s agent.'

   <limitedRightToConfirm> xsd:boolean </limitedRightToConfirm> [0..1]
   'Has the meaning defined as part of the 1997 ISDA Government Bond Option Definitions, section 4.5 Limited Right to Confirm Exercise. If present,
   (i) the Seller may request the Buyer to confirm its intent if not done on or before the expiration time on the Expiration date (ii) specific rules
   will apply in relation to the settlement mode.'

   <splitTicket> xsd:boolean </splitTicket> [0..1]
   'Typically applicable to the physical settlement of bond and convertible bond options. If present, means that the Party required to deliver the
   bonds will divide those to be delivered as notifying party desires to facilitate delivery obligations.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ExerciseProcedure">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="manualExercise" type=" ManualExercise "/>
      <xsd:element name="automaticExercise" type=" AutomaticExercise "/>
    </xsd:choice>
    <xsd:element name="followUpConfirmation" type=" xsd:boolean "/>
    <xsd:element name="limitedRightToConfirm" type=" xsd:boolean " minOccurs="0"/>
    <xsd:element name="splitTicket" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingRate

[Table of contents]

Super-types: Rate < FloatingRate (by extension)

Sub-types:

- FloatingRateCalculation (by extension)

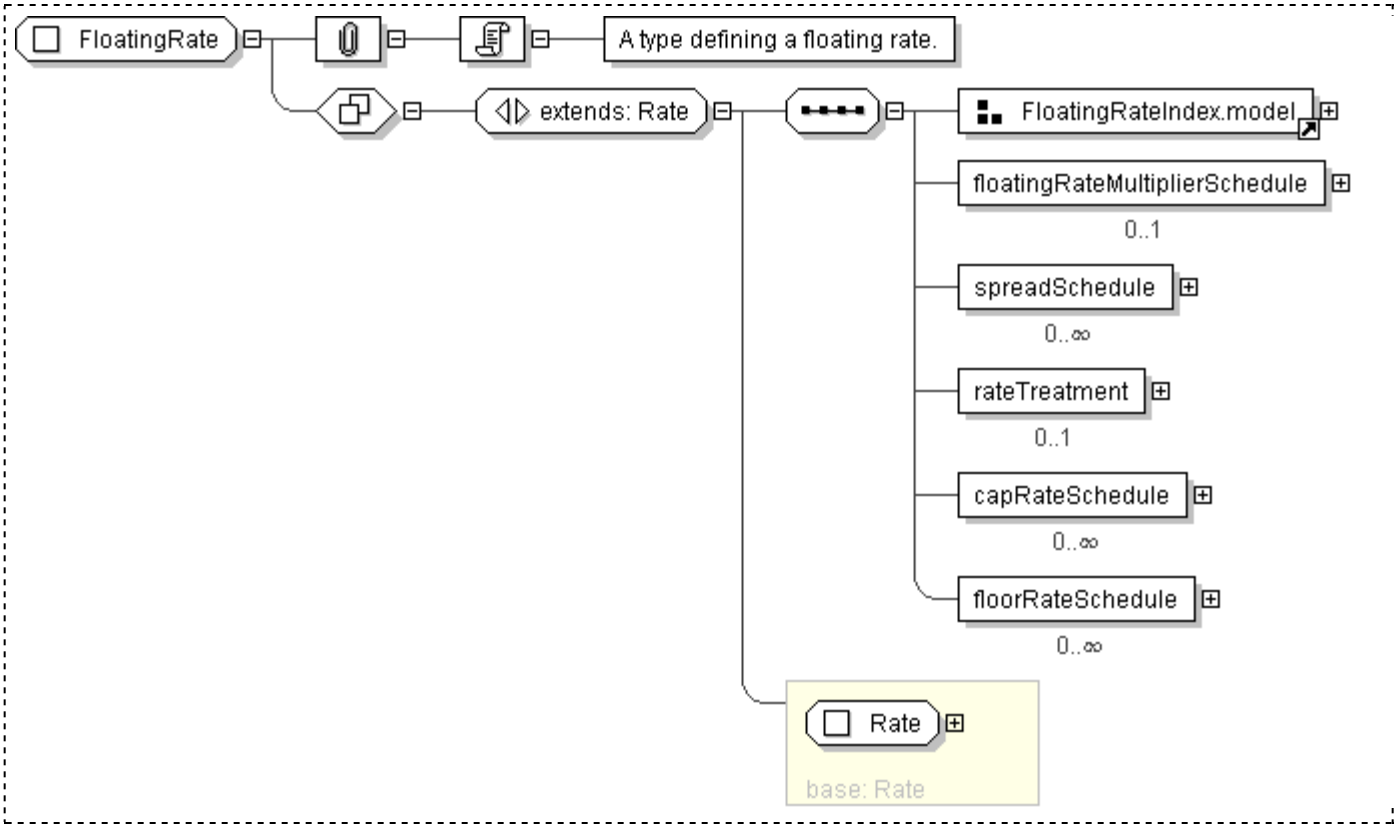
Name	FloatingRate
Used by (from the same schema document)	Complex Type StubValue
Abstract	no
Documentation	A type defining a floating rate.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
    'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <floatingRateMultiplierSchedule> Schedule </floatingRateMultiplierSchedule> [0..1]
    'A rate multiplier or multiplier schedule to apply to the floating rate. A
    multiplier schedule is expressed as explicit multipliers and dates. In the case
    of a schedule, the step dates may be subject to adjustment in accordance with
    any adjustments specified in the calculationPeriodDatesAdjustments. The
    multiplier can be a positive or negative decimal. This element should only be
    included if the multiplier is not equal to 1 (one) for the term of the stream.'

   <spreadSchedule> SpreadSchedule </spreadSchedule> [0..*]
    'The ISDA Spread or a Spread schedule expressed as explicit spreads and dates.
    In the case of a schedule, the step dates may be subject to adjustment in
    accordance with any adjustments specified in calculationPeriodDatesAdjustments.
    The spread is a per annum rate, expressed as a decimal. For purposes of
    determining a calculation period amount, if positive the spread will be added
    to the floating rate and if negative the spread will be subtracted from the
    floating rate. A positive 10 basis point (0.1%) spread would be represented as
    0.001.'

   <rateTreatment> RateTreatmentEnum </rateTreatment> [0..1]
    'The specification of any rate conversion which needs to be applied to the
    observed rate before being used in any calculations. The two common conversions
    are for securities quoted on a bank discount basis which will need to be
    converted to either a Money Market Yield or Bond Equivalent Yield. See the
    Annex to the 2000 ISDA Definitions, Section 7.3. Certain General Definitions
    Relating to Floating Rate Options, paragraphs (g) and (h) for definitions of
    these terms.'

   <capRateSchedule> StrikeSchedule </capRateSchedule> [0..*]
    'The cap rate or cap rate schedule, if any, which applies to the floating rate.
    The cap rate (strike) is only required where the floating rate on a swap stream
    is capped at a certain level. A cap rate schedule is expressed as explicit cap
    rates and dates and the step dates may be subject to adjustment in accordance
    with any adjustments specified in calculationPeriodDatesAdjustments. The cap
    rate is assumed to be exclusive of any spread and is a per annum rate,
    expressed as a decimal. A cap rate of 5% would be represented as 0.05.'

   <floorRateSchedule> StrikeSchedule </floorRateSchedule> [0..*]
    'The floor rate or floor rate schedule, if any, which applies to the floating
```

```
rate. The floor rate (strike) is only required where the floating rate on a
swap stream is floored at a certain strike level. A floor rate schedule is
expressed as explicit floor rates and dates and the step dates may be subject
to adjustment in accordance with any adjustments specified in
calculationPeriodDatesAdjustments. The floor rate is assumed to be exclusive of
any spread and is a per annum rate, expressed as a decimal. A floor rate of 5%
would be represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingRate">
   <xsd:complexContent>
      <xsd:extension base=" Rate ">
         <xsd:sequence>
            <xsd:group ref=" FloatingRateIndex.model "/>
            <xsd:element name="floatingRateMultiplierSchedule" type=" Schedule "
            minOccurs="0"/>
            <xsd:element name="spreadSchedule" type=" SpreadSchedule " minOccurs="0"
            maxOccurs="unbounded"/>
            <xsd:element name="rateTreatment" type=" RateTreatmentEnum " minOccurs="0"/>
            <xsd:element name="capRateSchedule" type=" StrikeSchedule " minOccurs="0"
            maxOccurs="unbounded"/>
            <xsd:element name="floorRateSchedule" type=" StrikeSchedule " minOccurs="0"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingRateCalculation

[Table of contents]

Super-types:	Rate < FloatingRate (by extension) < FloatingRateCalculation (by extension)
Sub-types:	None

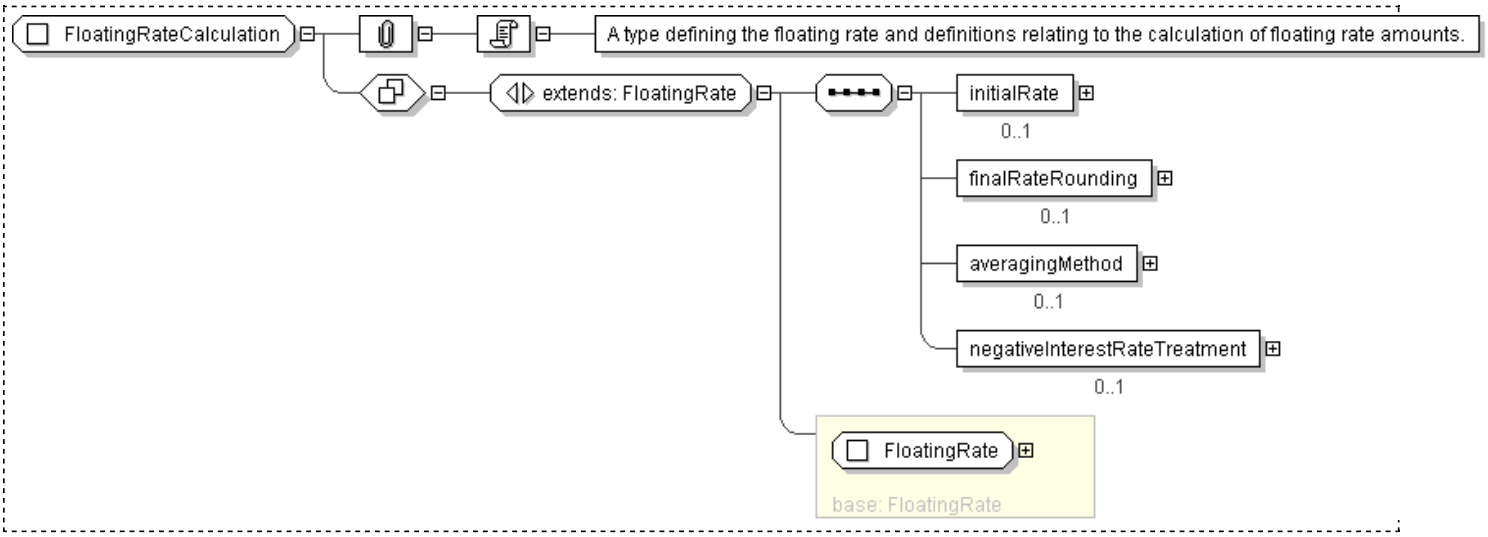
Name	FloatingRateCalculation
Used by (from the same schema document)	Complex Type InterestAccrualsMethod
Abstract	no
Documentation	A type defining the floating rate and definitions relating to the calculation of floating rate amounts.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   <indexTenor> Period </indexTenor> [0..1]
    'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

   <floatingRateMultiplierSchedule> Schedule </floatingRateMultiplierSchedule> [0..1]
    'A rate multiplier or multiplier schedule to apply to the floating rate. A multiplier schedule is
    expressed as explicit multipliers and dates. In the case of a schedule, the step dates may be
    subject to adjustment in accordance with any adjustments specified in the
    calculationPeriodDatesAdjustments. The multiplier can be a positive or negative decimal. This
    element should only be included if the multiplier is not equal to 1 (one) for the term of the
    stream.'

   <spreadSchedule> SpreadSchedule </spreadSchedule> [0..*]
    'The ISDA Spread or a Spread schedule expressed as explicit spreads and dates. In the case of a
    schedule, the step dates may be subject to adjustment in accordance with any adjustments specified
    in calculationPeriodDatesAdjustments. The spread is a per annum rate, expressed as a decimal. For
    purposes of determining a calculation period amount, if positive the spread will be added to the
    floating rate and if negative the spread will be subtracted from the floating rate. A positive 10
    basis point (0.1%) spread would be represented as 0.001.'

   <rateTreatment> RateTreatmentEnum </rateTreatment> [0..1]
    'The specification of any rate conversion which needs to be applied to the observed rate before
    being used in any calculations. The two common conversions are for securities quoted on a bank
    discount basis which will need to be converted to either a Money Market Yield or Bond Equivalent
    Yield. See the Annex to the 2000 ISDA Definitions, Section 7.3. Certain General Definitions
    Relating to Floating Rate Options, paragraphs (g) and (h) for definitions of these terms.'

   <capRateSchedule> StrikeSchedule </capRateSchedule> [0..*]
    'The cap rate or cap rate schedule, if any, which applies to the floating rate. The cap rate
    (strike) is only required where the floating rate on a swap stream is capped at a certain level. A
    cap rate schedule is expressed as explicit cap rates and dates and the step dates may be subject to
    adjustment in accordance with any adjustments specified in calculationPeriodDatesAdjustments. The
    cap rate is assumed to be exclusive of any spread and is a per annum rate, expressed as a decimal.
    A cap rate of 5% would be represented as 0.05.'

   <floorRateSchedule> StrikeSchedule </floorRateSchedule> [0..*]
    'The floor rate or floor rate schedule, if any, which applies to the floating rate. The floor rate
    (strike) is only required where the floating rate on a swap stream is floored at a certain strike
    level. A floor rate schedule is expressed as explicit floor rates and dates and the step dates may
    be subject to adjustment in accordance with any adjustments specified in
    calculationPeriodDatesAdjustments. The floor rate is assumed to be exclusive of any spread and is a
    per annum rate, expressed as a decimal. A floor rate of 5% would be represented as 0.05.'

   <initialRate> xsd:decimal </initialRate> [0..1]
    'The initial floating rate reset agreed between the principal parties involved in the trade. This
    is assumed to be the first required reset rate for the first regular calculation period. It should
    only be included when the rate is not equal to the rate published on the source implied by the
    floating rate index. An initial rate of 5% would be represented as 0.05.'

   <finalRateRounding> Rounding </finalRateRounding> [0..1]
    'The rounding convention to apply to the final rate used in determination of a calculation period
    amount.'

   <averagingMethod> AveragingMethodEnum </averagingMethod> [0..1]
    'If averaging is applicable, this component specifies whether a weighted or unweighted average
    method of calculation is to be used. The component must only be included when averaging applies.'

   <negativeInterestRateTreatment> NegativeInterestRateTreatmentEnum </negativeInterestRateTreatment> [0..1]
    'The specification of any provisions for calculating payment obligations when a floating rate is
    negative (either due to a quoted negative floating rate or by operation of a spread that is
    subtracted from the floating rate).'
```

```
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingRateCalculation">
  <xsd:complexContent>
    <xsd:extension base=" FloatingRate ">
      <xsd:sequence>
        <xsd:element name="initialRate" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="finalRateRounding" type=" Rounding " minOccurs="0"/>
        <xsd:element name="averagingMethod" type=" AveragingMethodEnum " minOccurs="0"/>
        <xsd:element name="negativeInterestRateTreatment" type=" NegativeInterestRateTreatmentEnum "
        minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FloatingRateIndex

[Table of contents]

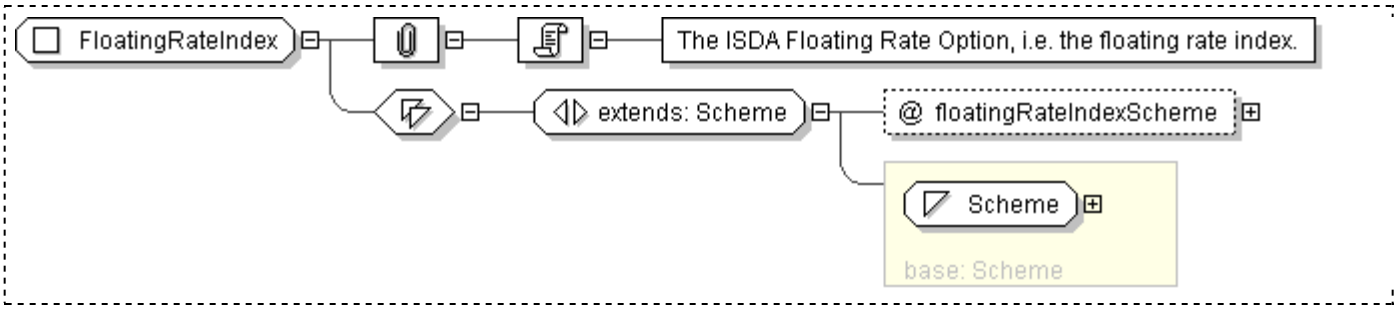
Super-types:	xsd:normalizedString < Scheme (by restriction) < FloatingRateIndex (by extension)
Sub-types:	None

Name	FloatingRateIndex
Used by (from the same schema document)	Complex Type ForecastRateIndex , Model Group FloatingRateIndex.model
Abstract	no
Documentation	The ISDA Floating Rate Option, i.e. the floating rate index.

XML Instance Representation

```
<...
 floatingRateIndexScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FloatingRateIndex">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="floatingRateIndexScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/floating-rate-index"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ForecastRateIndex

[Table of contents]

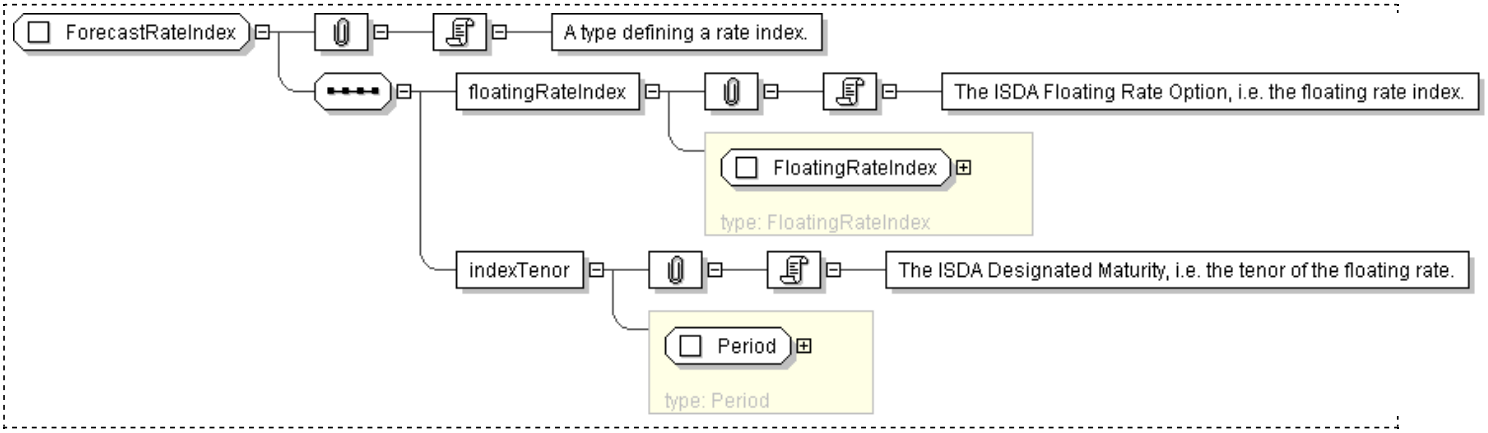
Super-types:	None
Sub-types:	None

Name	ForecastRateIndex
Abstract	no
Documentation	A type defining a rate index.

XML Instance Representation

```
<...>
   <floatingRateIndex> FloatingRateIndex </floatingRateIndex> [1]
   'The ISDA Floating Rate Option, i.e. the floating rate index.'

   <indexTenor> Period </indexTenor> [1]
   'The ISDA Designated Maturity, i.e. the tenor of the floating rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ForecastRateIndex">
   <xsd:sequence>
      <xsd:element name="floatingRateIndex" type=" FloatingRateIndex "/>
      <xsd:element name="indexTenor" type=" Period "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Formula

[Table of contents]

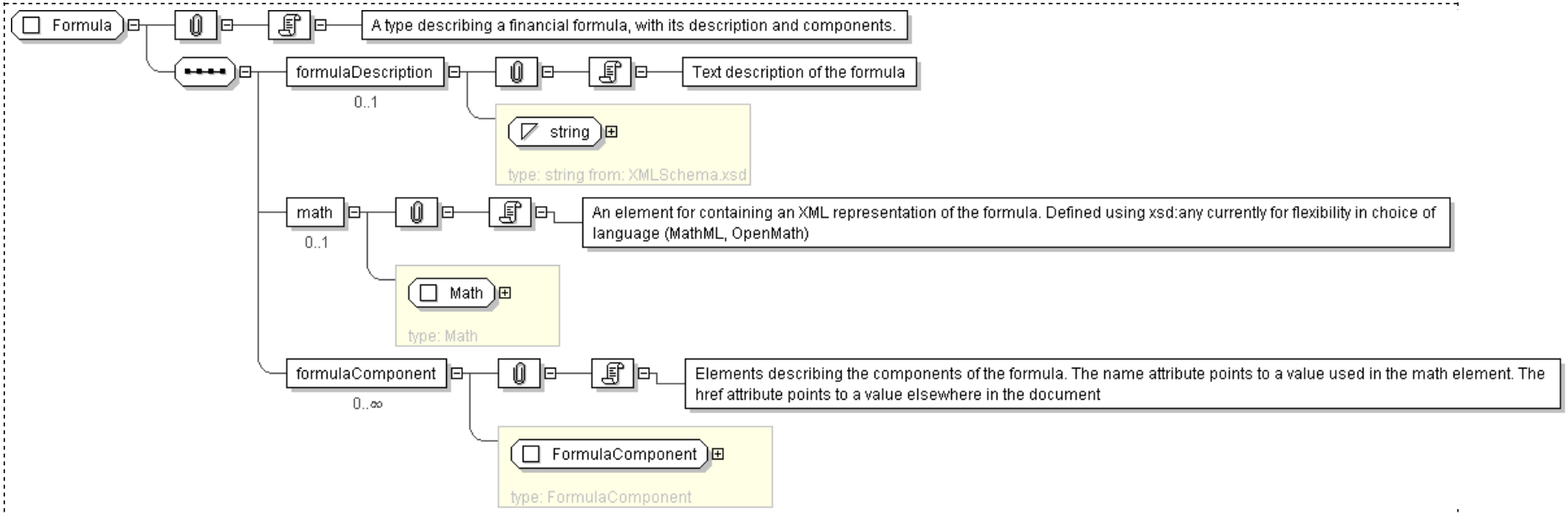
Super-types:	None
Sub-types:	None

Name	Formula
Used by (from the same schema document)	Complex Type FormulaComponent
Abstract	no
Documentation	A type describing a financial formula, with its description and components.

XML Instance Representation

```
<...>
  <formulaDescription> xsd:string </formulaDescription> [0..1]
  'Text description of the formula'

  <math> Math </math> [0..1]
  'An element for containing an XML representation of the formula. Defined using xsd:any currently for flexibility in choice of
  language (MathML, OpenMath)'

  <formulaComponent> FormulaComponent </formulaComponent> [0..*]
  'Elements describing the components of the formula. The name attribute points to a value used in the math element. The href attribute
  points to a value elsewhere in the document'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Formula">
   <xsd:sequence>
      <xsd:element name="formulaDescription" type=" xsd:string " minOccurs="0"/>
      <xsd:element name="math" type=" Math " minOccurs="0"/>
      <xsd:element name="formulaComponent" type=" FormulaComponent " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FormulaComponent

[Table of contents]

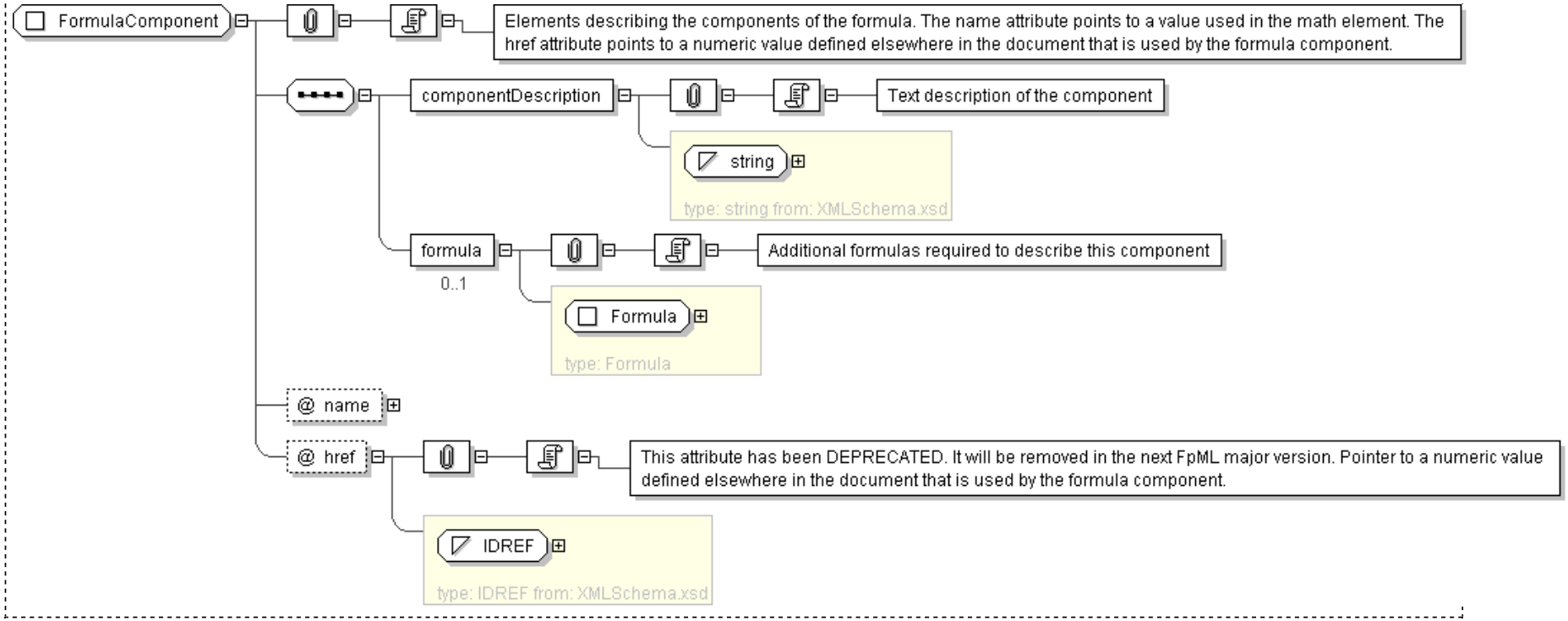
Super-types:	None
Sub-types:	None

Name	FormulaComponent
Used by (from the same schema document)	Complex Type Formula
Abstract	no
Documentation	Elements describing the components of the formula. The name attribute points to a value used in the math element. The href attribute points to a numeric value defined elsewhere in the document that is used by the formula component.

XML Instance Representation

```
<...
 name=" xsd:normalizedString [0..1]"
 href=" xsd:IDREF [0..1]
 'This attribute has been DEPRECATED. It will be removed in the next FpML major version. Pointer to a numeric value defined
 elsewhere in the document that is used by the formula component.'

 ">
   <componentDescription> xsd:string </componentDescription> [1]
   'Text description of the component'

   <formula> Formula </formula> [0..1]
   'Additional formulas required to describe this component'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FormulaComponent">
  <xsd:sequence>
    <xsd:element name="componentDescription" type=" xsd:string "/>
    <xsd:element name="formula" type=" Formula " minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="name" type=" xsd:normalizedString "/>
  <xsd:attribute name="href" type=" xsd:IDREF " deprecated="true" deprecatedReason="There was no definition on where this
  attribute should point at and no clear usage."/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Frequency

[Table of contents]

Super-types: None

Sub-types:

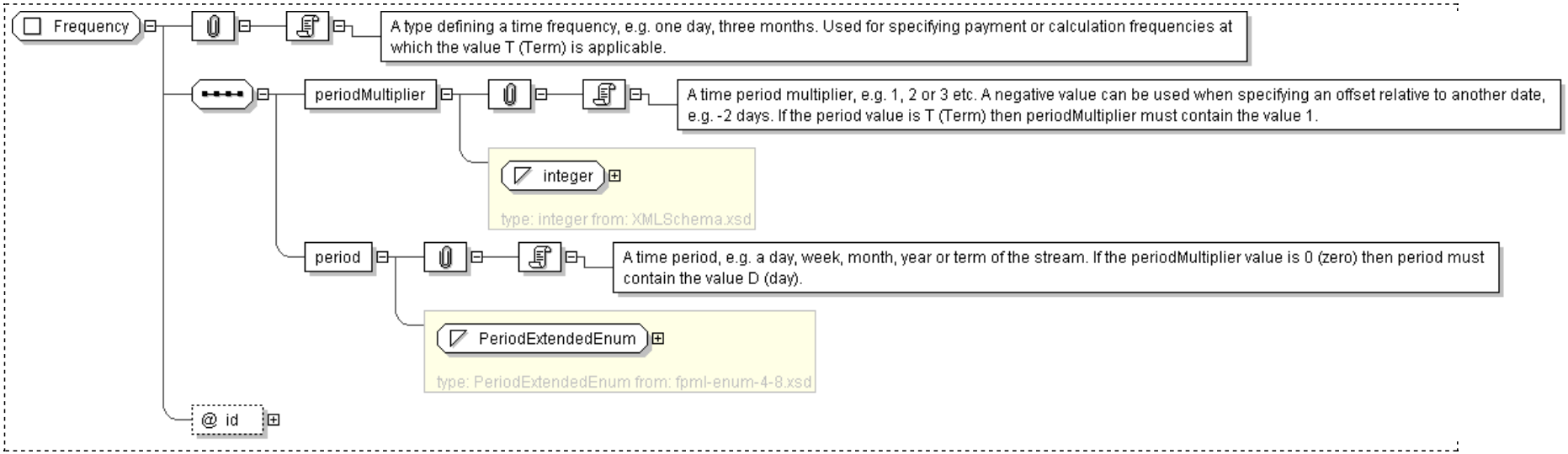
- CalculationPeriodFrequency (by extension)
- ResetFrequency (by extension)

Name	Frequency
Abstract	no
Documentation	A type defining a time frequency, e.g. one day, three months. Used for specifying payment or calculation frequencies at which the value T (Term) is applicable.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to another date,
   e.g. -2 days. If the period value is T (Term) then periodMultiplier must contain the value 1.'

   <period> PeriodExtendedEnum </period> [1]
   'A time period, e.g. a day, week, month, year or term of the stream. If the periodMultiplier value is 0 (zero) then period must
   contain the value D (day).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Frequency">
  <xsd:sequence>
    <xsd:element name="periodMultiplier" type=" xsd:integer "/>
    <xsd:element name="period" type=" PeriodExtendedEnum "/>
```

```
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FutureValueAmount

[Table of contents]

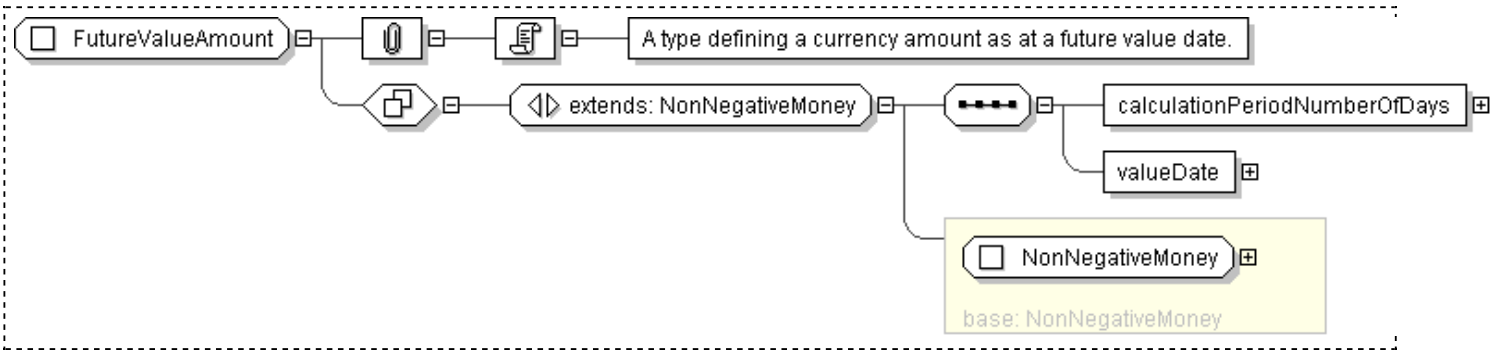
Super-types:	MoneyBase < NonNegativeMoney (by extension) < FutureValueAmount (by extension)
Sub-types:	None

Name	FutureValueAmount
Abstract	no
Documentation	A type defining a currency amount as at a future value date.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

   <amount> NonNegativeDecimal </amount> [1]
    'The non negative monetary quantity in currency units.'

   <calculationPeriodNumberOfDays> xsd:positiveInteger </calculationPeriodNumberOfDays> [1]
    'The number of days from the adjusted calculation period start date to the adjusted
    value date, calculated in accordance with the applicable day count fraction.'

   <valueDate> xsd:date </valueDate> [1]
    'Adjusted value date of the future value amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FutureValueAmount">
  <xsd:complexContent>
    <xsd:extension base=" NonNegativeMoney ">
      <xsd:sequence>
        <xsd:element name="calculationPeriodNumberOfDays" type=" xsd:positiveInteger "/>
        <xsd:element name="valueDate" type=" xsd:date "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxCashSettlement

[Table of contents]

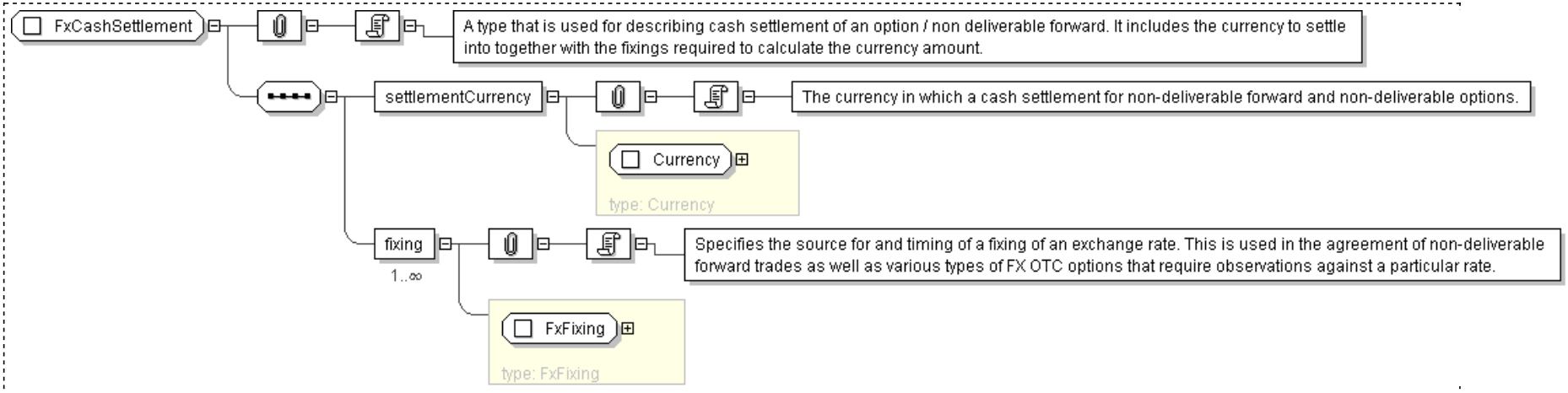
Super-types:	None
Sub-types:	None

Name	FxCashSettlement
Abstract	no
Documentation	A type that is used for describing cash settlement of an option / non deliverable forward. It includes the currency to settle into together with the fixings required to calculate the currency amount.

XML Instance Representation

```
<...>
   <settlementCurrency> Currency </settlementCurrency> [1]
   'The currency in which a cash settlement for non-deliverable forward and non-deliverable options.'

   <fixing> FxFixing </fixing> [1..*]
   'Specifies the source for and timing of a fixing of an exchange rate. This is used in the agreement of non-deliverable forward
   trades as well as various types of FX OTC options that require observations against a particular rate.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxCashSettlement">
   <xsd:sequence>
      <xsd:element name="settlementCurrency" type=" Currency "/>
      <xsd:element name="fixing" type=" FxFixing " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxFixing

[Table of contents]

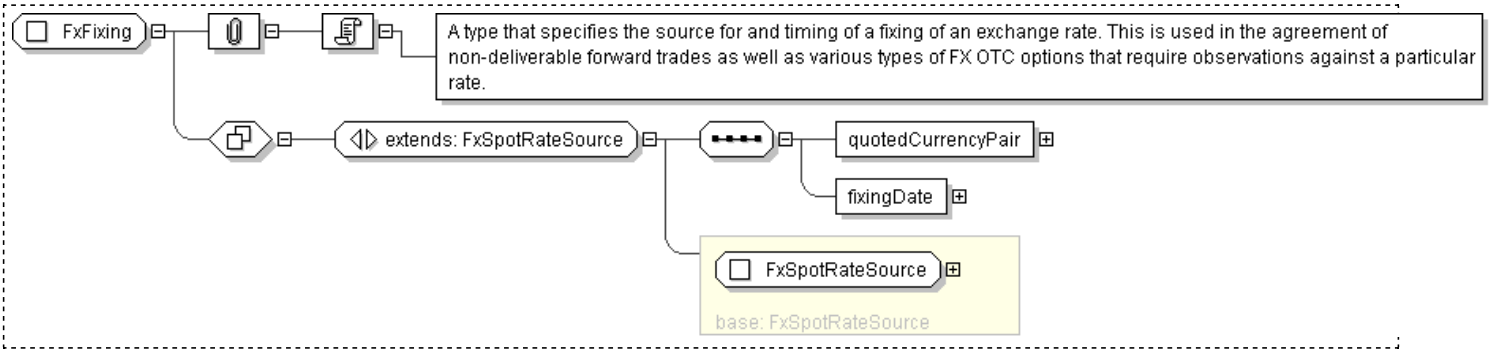
Super-types:	FxSpotRateSource < FxFixing (by extension)
Sub-types:	None

Name	FxFixing
Used by (from the same schema document)	Complex Type FxCashSettlement
Abstract	no
Documentation	A type that specifies the source for and timing of a fixing of an exchange rate. This is used in the agreement of non-deliverable forward trades as well as various types of FX OTC options that require observations against a particular rate.

XML Instance Representation

```
<...>
  <primaryRateSource> InformationSource </primaryRateSource> [1]
  'The primary source for where the rate observation will occur. Will typically be either a page or a
  reference bank published rate.'

  <secondaryRateSource> InformationSource </secondaryRateSource> [0..1]
  'An alternative, or secondary, source for where the rate observation will occur. Will typically be
  either a page or a reference bank published rate.'

  <fixingTime> BusinessCenterTime </fixingTime> [1]
  'The time at which the spot currency exchange rate will be observed. It is specified as a time in a
  specific business center, e.g. 11:00am London time.'

  <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
  'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

  <fixingDate> xsd:date </fixingDate> [1]
  'Describes the specific date when a non-deliverable forward or non-deliverable option will \"fix\"
  against a particular rate, which will be used to compute the ultimate cash settlement.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxFixing">
  <xsd:complexContent>
    <xsd:extension base=" FxSpotRateSource ">
      <xsd:sequence>
        <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
        <xsd:element name="fixingDate" type=" xsd:date "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxRate

[Table of contents]

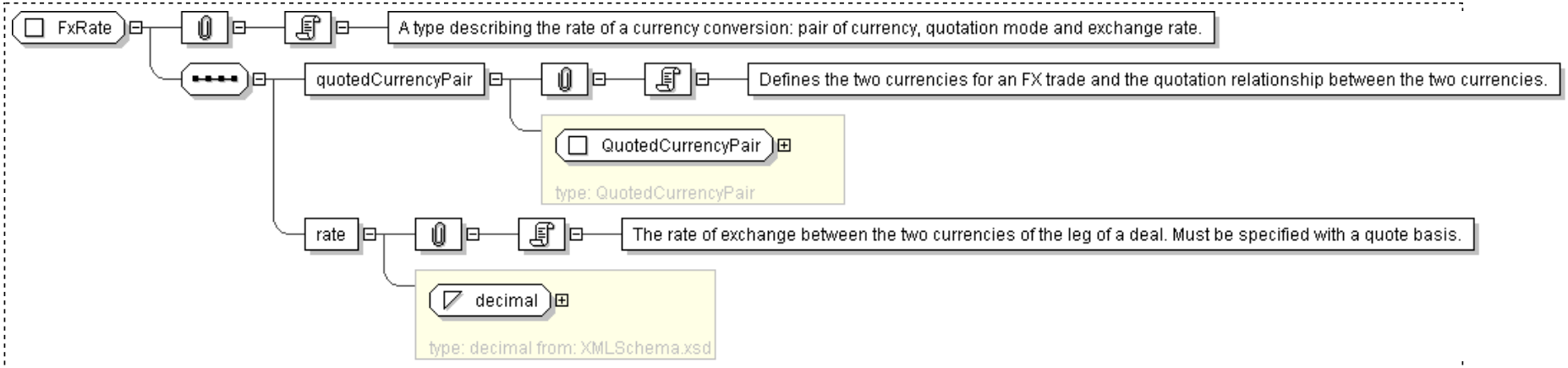
Super-types:	None
Sub-types:	None

Name	FxRate
Abstract	no
Documentation	A type describing the rate of a currency conversion: pair of currency, quotation mode and exchange rate.

XML Instance Representation

```
<...>
   <quotedCurrencyPair> QuotedCurrencyPair </quotedCurrencyPair> [1]
   'Defines the two currencies for an FX trade and the quotation relationship between the two currencies.'

   <rate> xsd:decimal </rate> [1]
   'The rate of exchange between the two currencies of the leg of a deal. Must be specified with a quote basis.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxRate">
   <xsd:sequence>
      <xsd:element name="quotedCurrencyPair" type=" QuotedCurrencyPair "/>
      <xsd:element name="rate" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: FxSpotRateSource

[Table of contents]

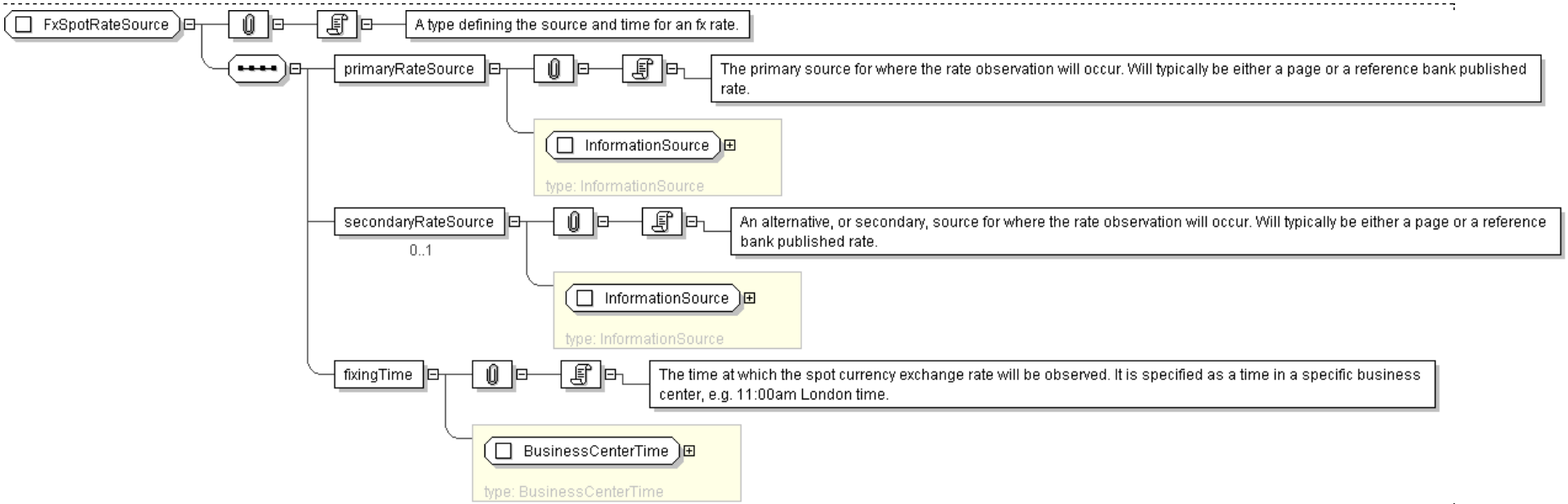
Super-types:	None
Sub-types:	• FxFixing (by extension)

Name	FxSpotRateSource
Abstract	no
Documentation	A type defining the source and time for an fx rate.

XML Instance Representation

```
<...>
   <primaryRateSource> InformationSource </primaryRateSource> [1]
   'The primary source for where the rate observation will occur. Will typically be either a page or a reference bank published rate.'

   <secondaryRateSource> InformationSource </secondaryRateSource> [0..1]
   'An alternative, or secondary, source for where the rate observation will occur. Will typically be either a page or a reference bank
   published rate.'

   <fixingTime> BusinessCenterTime </fixingTime> [1]
   'The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g. 11:00am
   London time.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="FxSpotRateSource">
   <xsd:sequence>
      <xsd:element name="primaryRateSource" type=" InformationSource "/>
      <xsd:element name="secondaryRateSource" type=" InformationSource " minOccurs="0"/>
```

```
    <xsd:element name="fixingTime" type=" BusinessCenterTime "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: GoverningLaw

[Table of contents]

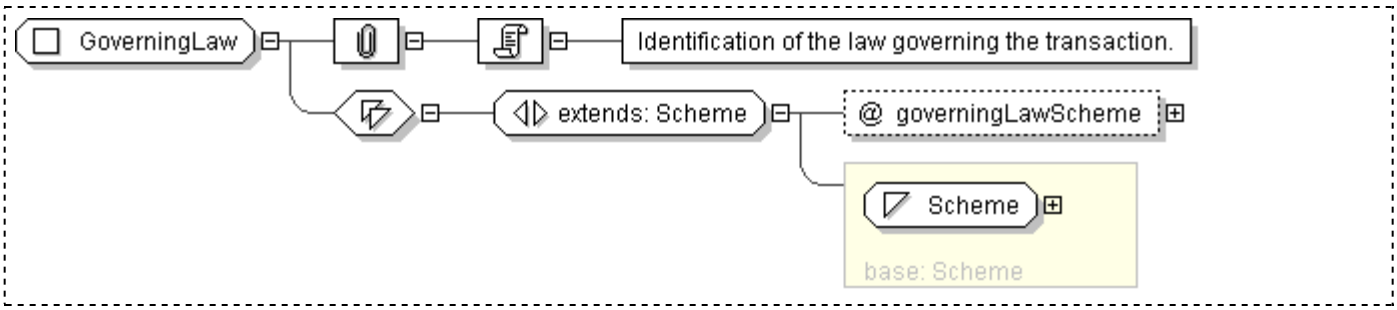
Super-types:	xsd:normalizedString < Scheme (by restriction) < GoverningLaw (by extension)
Sub-types:	None

Name	GoverningLaw
Abstract	no
Documentation	Identification of the law governing the transaction.

XML Instance Representation

```
<...
 governingLawScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="GoverningLaw">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="governingLawScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/governing-law"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

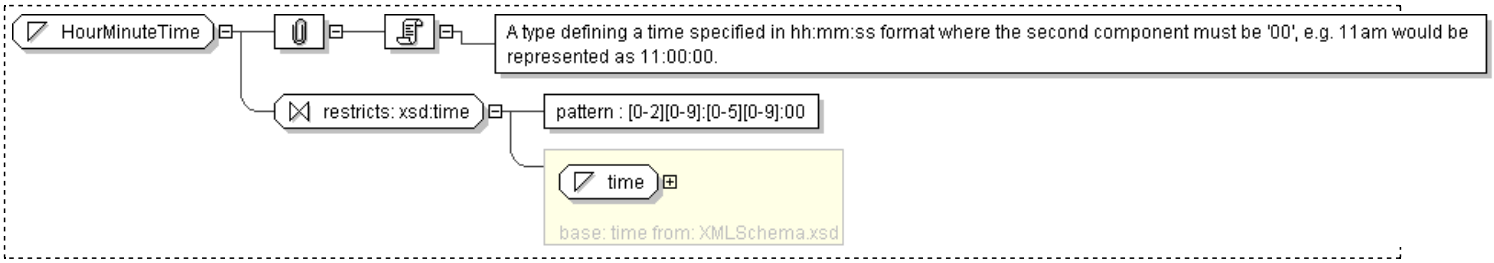
Simple Type: HourMinuteTime

[Table of contents]

Super-types:	xsd:time < HourMinuteTime (by restriction)
Sub-types:	None

Name	HourMinuteTime
Used by (from the same schema document)	Complex Type BusinessCenterTime , Complex Type PrevailingTime
Content	• Base XSD Type: time • *pattern* = [0-2][0-9]:[0-5][0-9]:00
Documentation	A type defining a time specified in hh:mm:ss format where the second component must be '00', e.g. 11am would be represented as 11:00:00.

Diagram



Schema Component Representation

```
<xsd:simpleType name="HourMinuteTime">
  <xsd:restriction base=" xsd:time ">
    <xsd:pattern value="[0-2][0-9]:[0-5][0-9]:00"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IdentifiedCurrency

[Table of contents]

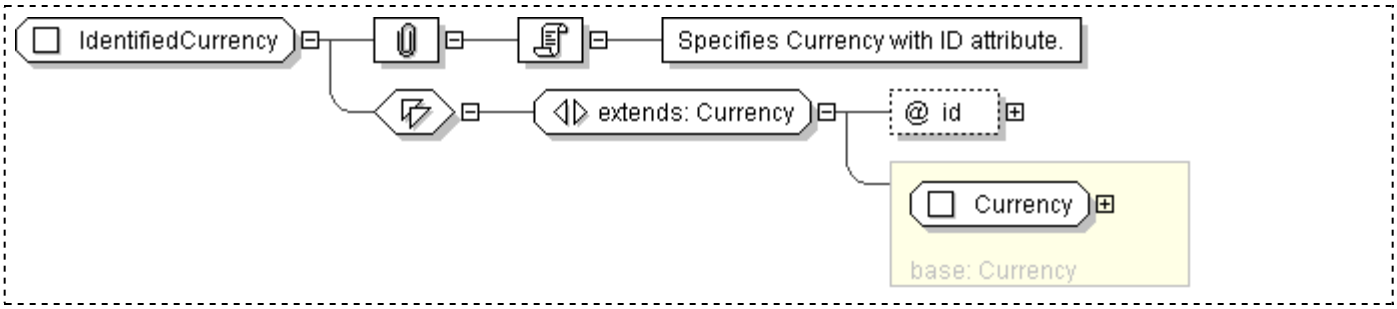
Super-types:	xsd:normalizedString < Scheme (by restriction) < Currency (by extension) < IdentifiedCurrency (by extension)
Sub-types:	None

Name	IdentifiedCurrency
Abstract	no
Documentation	Specifies Currency with ID attribute.

XML Instance Representation

```
<...
 currencyScheme=" xsd:anyURI [0..1]"
 id=" xsd:ID [0..1]">
   Currency
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IdentifiedCurrency">
   <xsd:simpleContent>
      <xsd:extension base=" Currency ">
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IdentifiedCurrencyReference

[Table of contents]

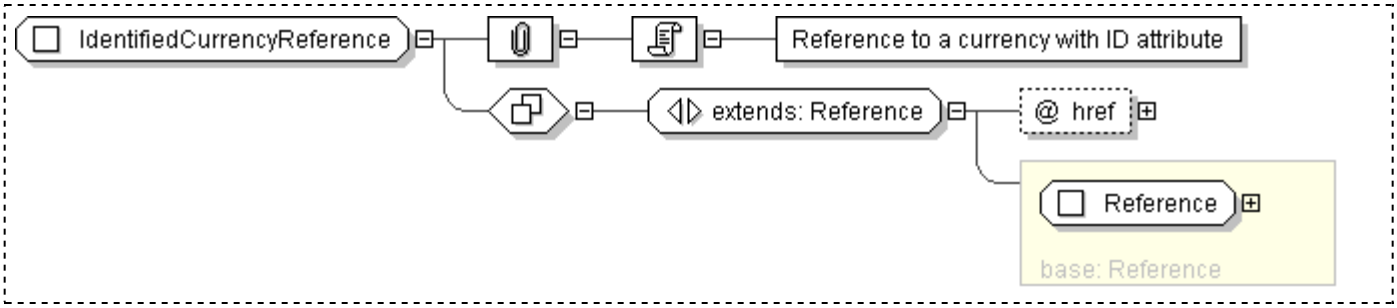
Super-types:	Reference < IdentifiedCurrencyReference (by extension)
Sub-types:	None

Name	IdentifiedCurrencyReference
Abstract	no
Documentation	Reference to a currency with ID attribute

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IdentifiedCurrencyReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="IdentifiedCurrency"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IdentifiedDate

[Table of contents]

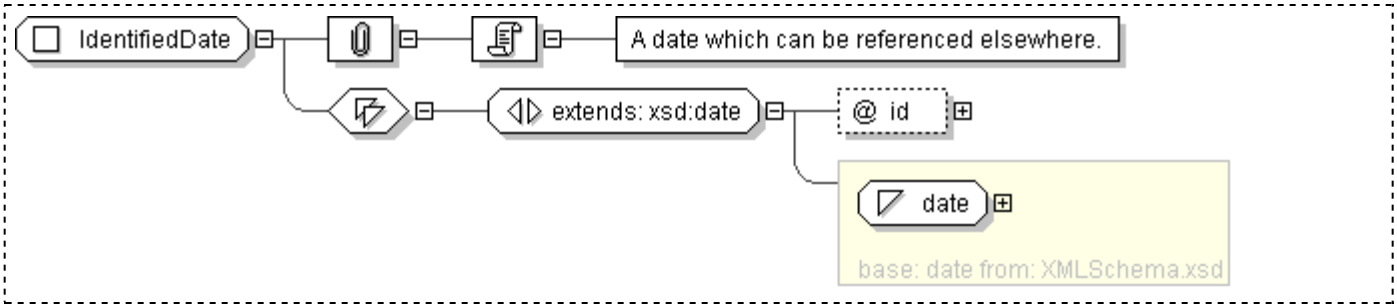
Super-types:	xsd:date < IdentifiedDate (by extension)
Sub-types:	None

Name	IdentifiedDate
Used by (from the same schema document)	Complex Type AdjustableDate , Complex Type AdjustableDate2 , Complex Type AdjustableDates , Complex Type AdjustableOrRelativeAndAdjustedDate , Complex Type Payment , Model Group VersionHistory.model
Abstract	no
Documentation	A date which can be referenced elsewhere.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   xsd:date
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IdentifiedDate">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:date ">
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IdentifiedPayerReceiver

[Table of contents]

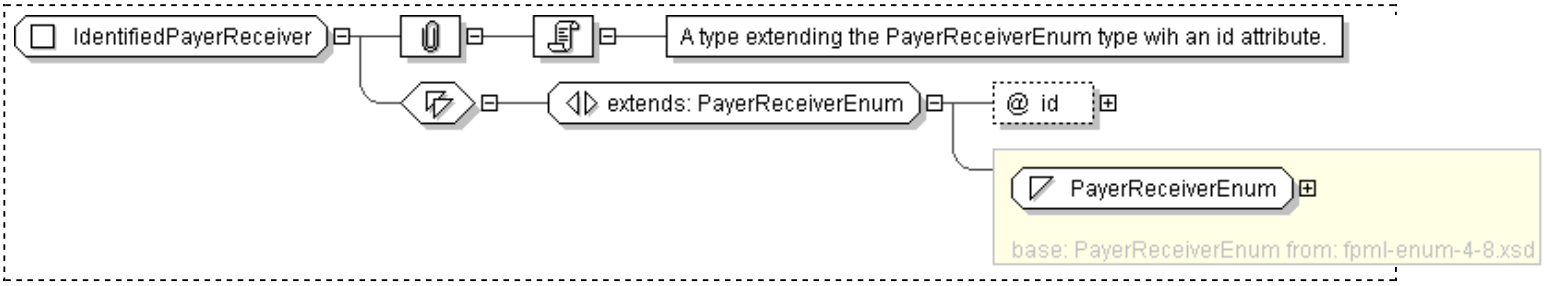
Super-types:	PayerReceiverEnum < IdentifiedPayerReceiver (by extension)
Sub-types:	None

Name	IdentifiedPayerReceiver
Used by (from the same schema document)	Complex Type Strike , Complex Type Strike , Complex Type StrikeSchedule , Complex Type StrikeSchedule
Abstract	no
Documentation	A type extending the PayerReceiverEnum type wih an id attribute.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   PayerReceiverEnum
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IdentifiedPayerReceiver">
   <xsd:simpleContent>
      <xsd:extension base=" PayerReceiverEnum ">
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InformationProvider

[Table of contents]

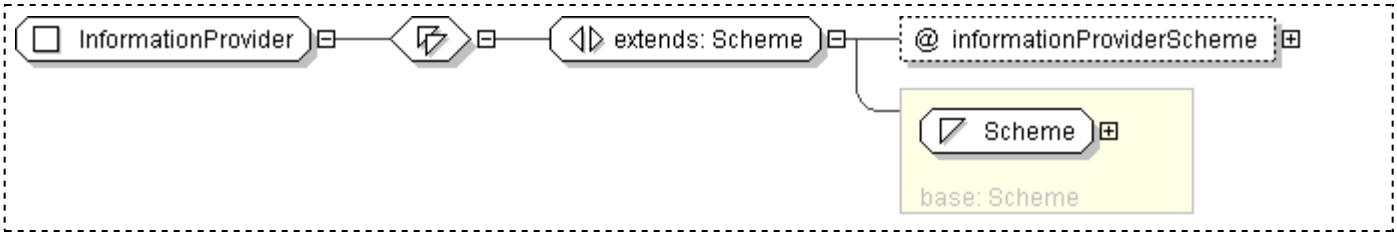
Super-types:	xsd:normalizedString < Scheme (by restriction) < InformationProvider (by extension)
Sub-types:	None

Name	InformationProvider
Used by (from the same schema document)	Complex Type InformationSource
Abstract	no

XML Instance Representation

```
<...
 informationProviderScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InformationProvider">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="informationProviderScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/information-provider"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InformationSource

[Table of contents]

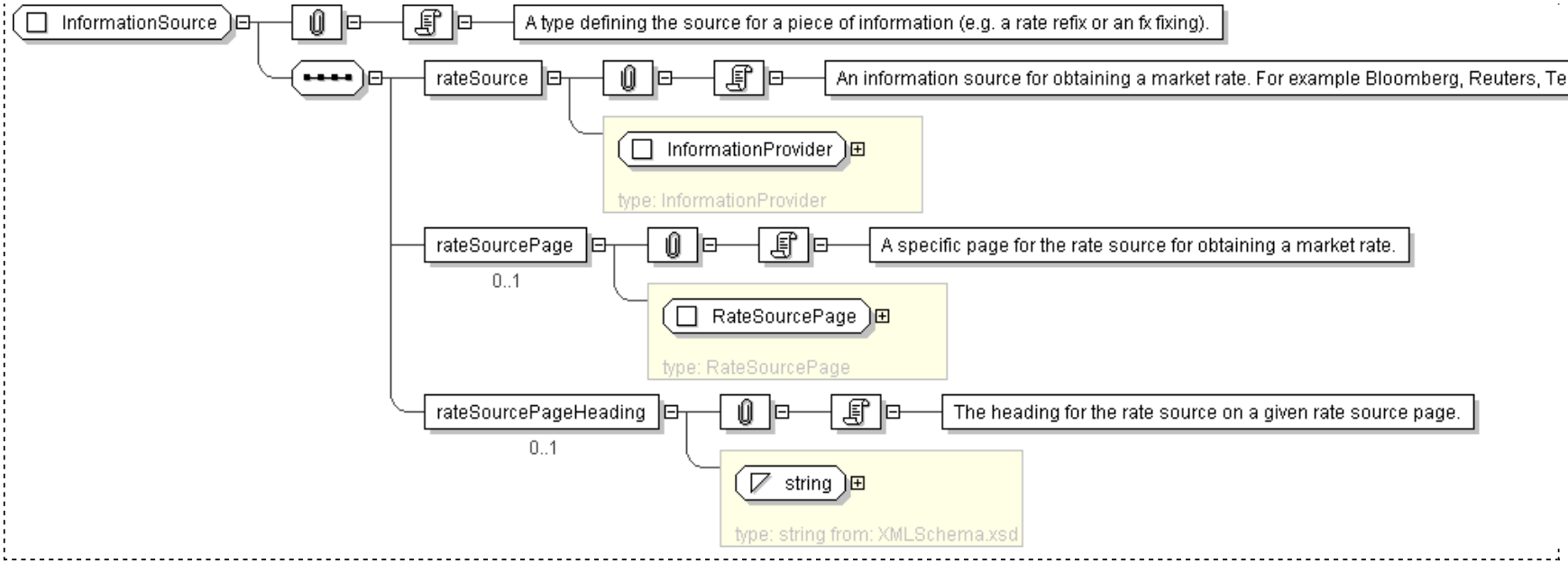
Super-types:	None
Sub-types:	None

Name	InformationSource
Used by (from the same schema document)	Complex Type FxSpotRateSource , Complex Type FxSpotRateSource , Complex Type SettlementRateSource
Abstract	no
Documentation	A type defining the source for a piece of information (e.g. a rate refix or an fx fixing).

XML Instance Representation

```
<...>
   <rateSource> InformationProvider </rateSource> [1]
   'An information source for obtaining a market rate. For example Bloomberg, Reuters, Telerate etc.'

   <rateSourcePage> RateSourcePage </rateSourcePage> [0..1]
   'A specific page for the rate source for obtaining a market rate.'

   <rateSourcePageHeading> xsd:string </rateSourcePageHeading> [0..1]
   'The heading for the rate source on a given rate source page.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InformationSource">
  <xsd:sequence>
    <xsd:element name="rateSource" type=" InformationProvider "/>
    <xsd:element name="rateSourcePage" type=" RateSourcePage " minOccurs="0"/>
    <xsd:element name="rateSourcePageHeading" type=" xsd:string " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InstrumentId

[Table of contents]

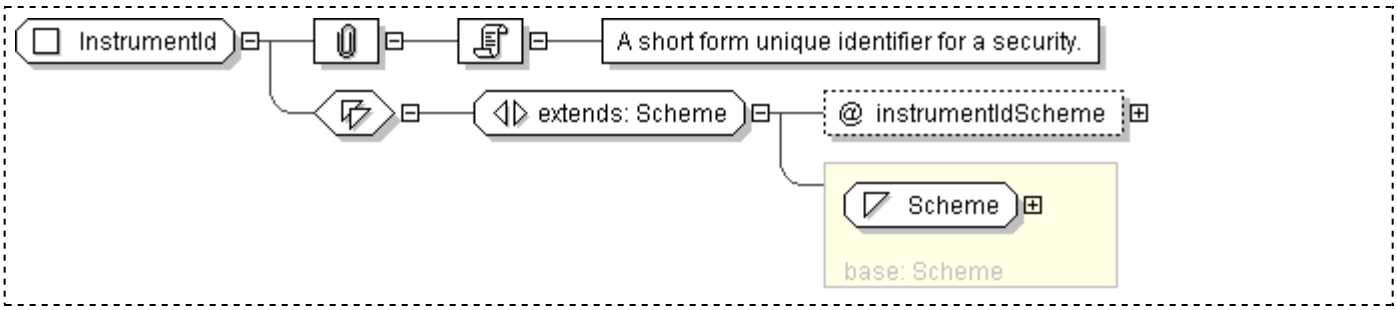
Super-types:	xsd:normalizedString < Scheme (by restriction) < InstrumentId (by extension)
Sub-types:	None

Name	InstrumentId
Abstract	no
Documentation	A short form unique identifier for a security.

XML Instance Representation

```
<...
 instrumentIdScheme=" xsd:anyURI [1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InstrumentId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="instrumentIdScheme" type=" xsd:anyURI " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestAccrualsCompoundingMethod

[Table of contents]

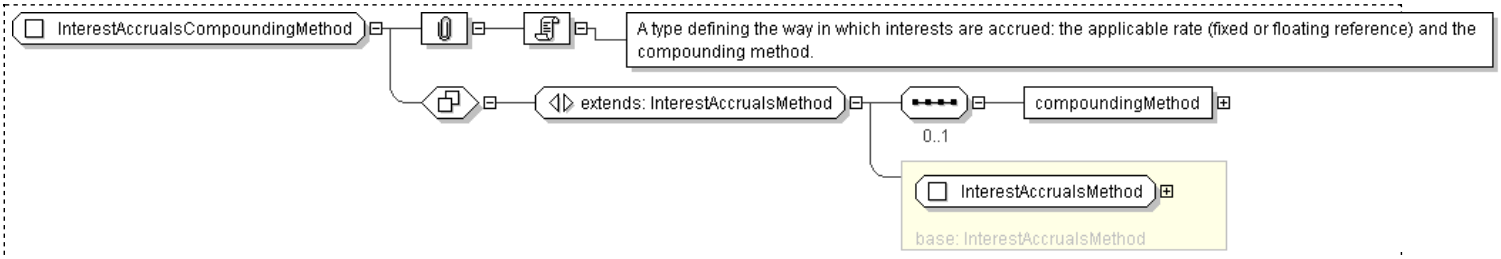
Super-types:	InterestAccrualsMethod < InterestAccrualsCompoundingMethod (by extension)
Sub-types:	None

Name	InterestAccrualsCompoundingMethod
Abstract	no
Documentation	A type defining the way in which interests are accrued: the applicable rate (fixed or floating reference) and the compounding method.

XML Instance Representation

```
<...>
  Start Choice [1]
    <floatingRateCalculation> FloatingRateCalculation </floatingRateCalculation> [1]
      'The floating rate calculation definitions'

    <fixedRate> xsd:decimal </fixedRate> [1]
      'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be
      represented as 0.05.'

  End Choice
  Start Sequence [0..1]
    <compoundingMethod> CompoundingMethodEnum </compoundingMethod> [1]
      'If more that one calculation period contributes to a single payment amount this element specifies whether
      compounding is applicable, and if so, what compounding method is to be used. This element must only be included
      when more that one calculation period contributes to a single payment amount.'

  End Sequence
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestAccrualsCompoundingMethod">
  <xsd:complexContent>
    <xsd:extension base=" InterestAccrualsMethod ">
      <xsd:sequence minOccurs="0">
        <xsd:element name="compoundingMethod" type=" CompoundingMethodEnum "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterestAccrualsMethod

[Table of contents]

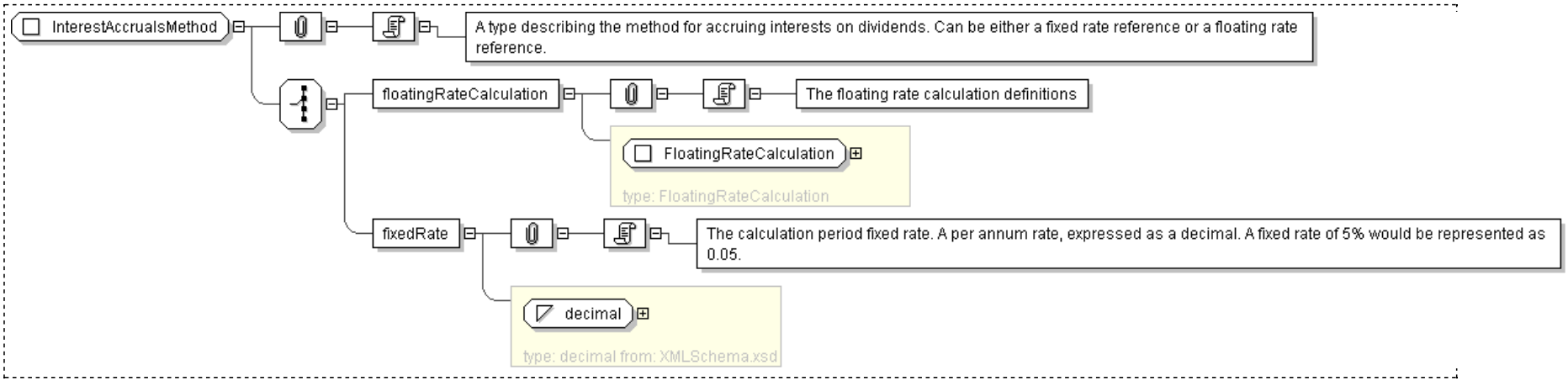
Super-types:	None
Sub-types:	• InterestAccrualsCompoundingMethod (by extension)

Name	InterestAccrualsMethod
Abstract	no
Documentation	A type describing the method for accruing interests on dividends. Can be either a fixed rate reference or a floating rate reference.

XML Instance Representation

```
<...>
  Start Choice [1]
    <floatingRateCalculation> FloatingRateCalculation </floatingRateCalculation> [1]
     'The floating rate calculation definitions'

    <fixedRate> xsd:decimal </fixedRate> [1]
     'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate of 5% would be represented as 0.05.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestAccrualsMethod">
  <xsd:choice>
    <xsd:element name="floatingRateCalculation" type=" FloatingRateCalculation "/>
    <xsd:element name="fixedRate" type=" xsd:decimal "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: IntermediaryInformation

[Table of contents]

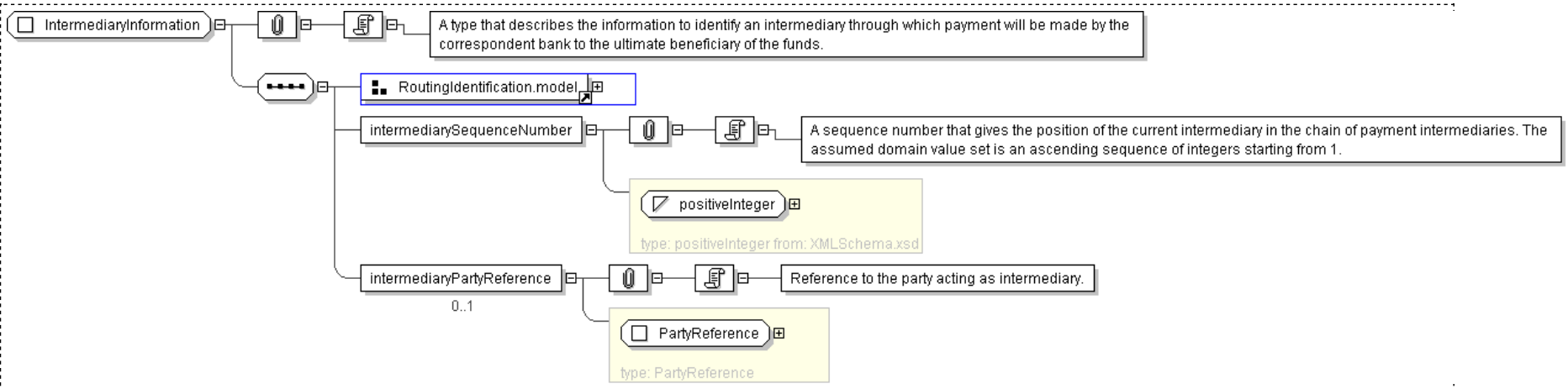
Super-types:	None
Sub-types:	None

Name	IntermediaryInformation
Used by (from the same schema document)	Complex Type SettlementInstruction
Abstract	no
Documentation	A type that describes the information to identify an intermediary through which payment will be made by the correspondent bank to the ultimate beneficiary of the funds.

XML Instance Representation

```
<...>
  Start Choice [1]
    <routingIds> RoutingIds </routingIds> [1]
     'A set of unique identifiers for a party, eachone identifying the party within a payment system. The assumption is that each party will not
     have more than one identifier within the same payment system.'

    <routingExplicitDetails> RoutingExplicitDetails </routingExplicitDetails> [1]
     'A set of details that is used to identify a party involved in the routing of a payment when the party does not have a code that identifies it
     within one of the recognized payment systems.'

    <routingIdsAndExplicitDetails> RoutingIdsAndExplicitDetails </routingIdsAndExplicitDetails> [1]
     'A combination of coded payment system identifiers and details for physical addressing for a party involved in the routing of a payment.'

  End Choice
  <intermediarySequenceNumber> xsd:positiveInteger </intermediarySequenceNumber> [1]
   'A sequence number that gives the position of the current intermediary in the chain of payment intermediaries. The assumed domain value set is an
   ascending sequence of integers starting from 1.'

  <intermediaryPartyReference> PartyReference </intermediaryPartyReference> [0..1]
   'Reference to the party acting as intermediary.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IntermediaryInformation">
  <xsd:sequence>
```

```
      <xsd:group ref=" RoutingIdentification.model "/>
      <xsd:element name="intermediarySequenceNumber" type=" xsd:positiveInteger "/>
      <xsd:element name="intermediaryPartyReference" type=" PartyReference " minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: InterpolationMethod

[Table of contents]

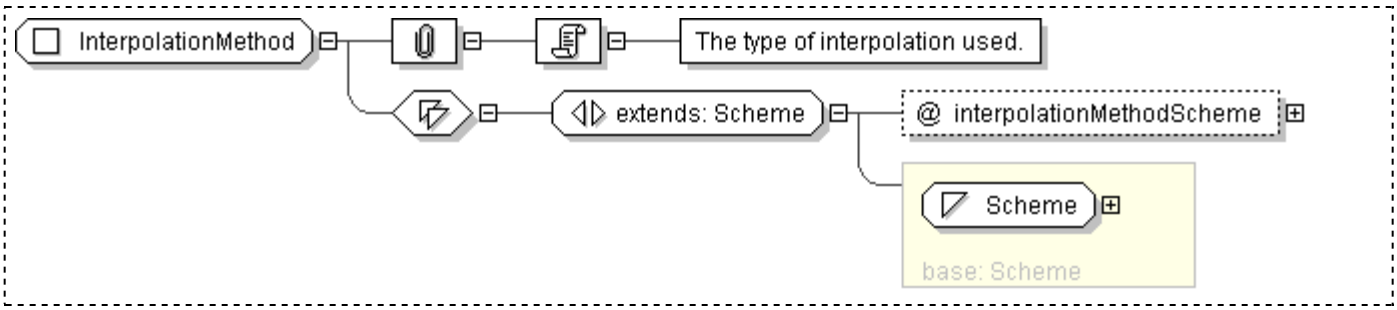
Super-types:	xsd:normalizedString < Scheme (by restriction) < InterpolationMethod (by extension)
Sub-types:	None

Name	InterpolationMethod
Abstract	no
Documentation	The type of interpolation used.

XML Instance Representation

```
<...
 interpolationMethodScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterpolationMethod">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="interpolationMethodScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/interpolation-method"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Leg

[Table of contents]

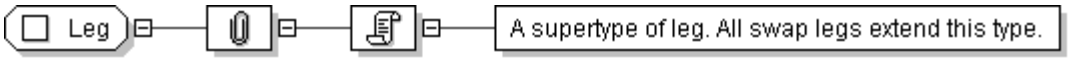
Super-types:	None
Sub-types:	None

Name	Leg
Abstract	yes
Documentation	A supertype of leg. All swap legs extend this type.

XML Instance Representation

```
<.../>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Leg" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LegalEntity

[Table of contents]

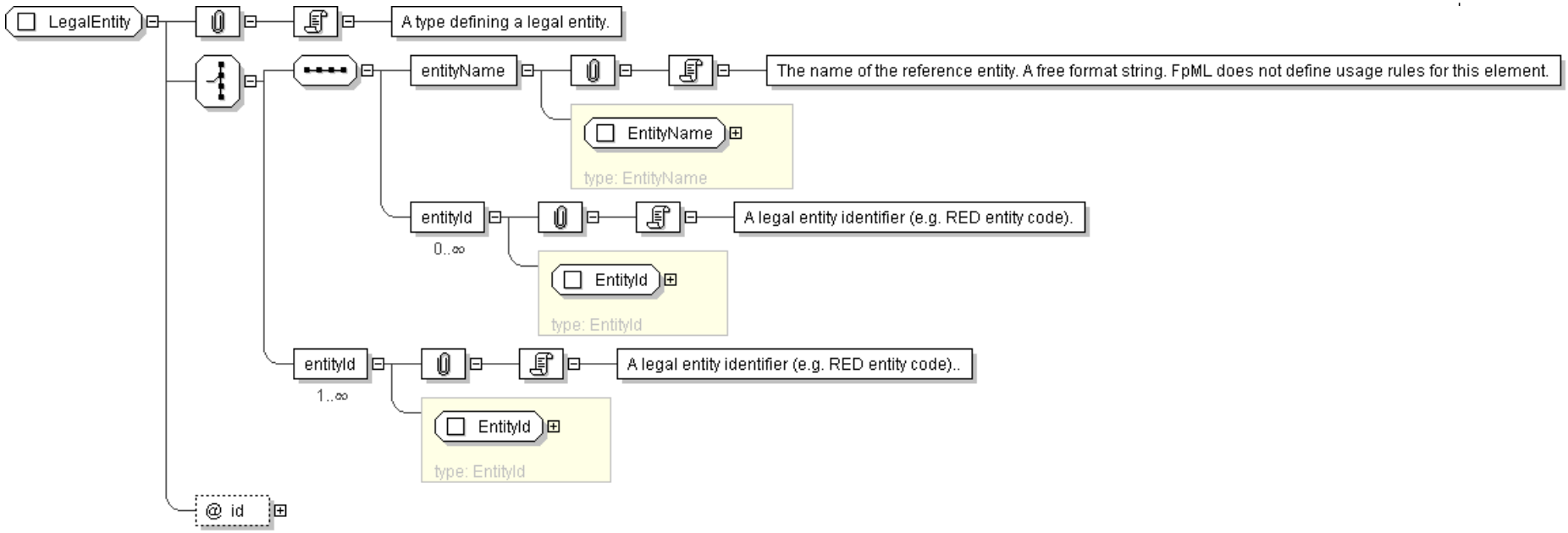
Super-types:	None
Sub-types:	None

Name	LegalEntity
Abstract	no
Documentation	A type defining a legal entity.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   Start Choice [1]
      <entityName> EntityName </entityName> [1]
      'The name of the reference entity. A free format string. FpML does not define usage rules for this element.'

      <entityId> EntityId </entityId> [0..*]
      'A legal entity identifier (e.g. RED entity code).'

      <entityId> EntityId </entityId> [1..*]
      'A legal entity identifier (e.g. RED entity code)..'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LegalEntity">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="entityName" type=" EntityName "/>
      <xsd:element name="entityId" type=" EntityId " minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:element name="entityId" type=" EntityId " maxOccurs="unbounded"/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: LegalEntityReference

[Table of contents]

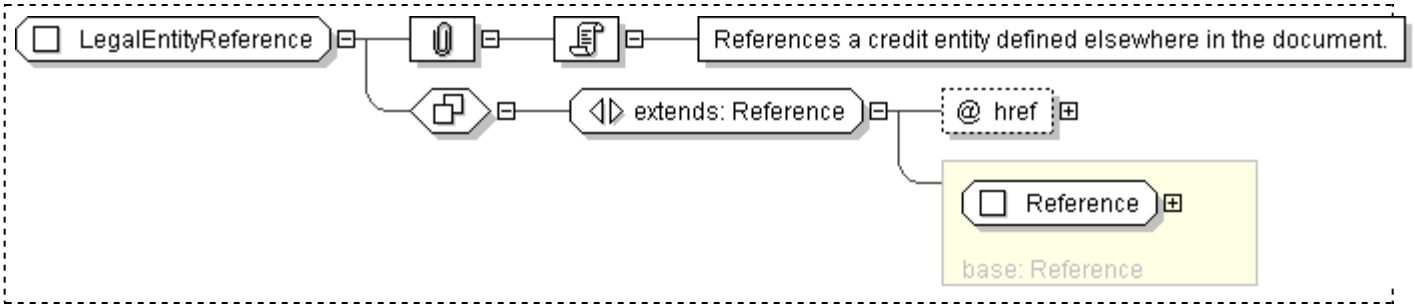
Super-types:	Reference < LegalEntityReference (by extension)
Sub-types:	None

Name	LegalEntityReference
Abstract	no
Documentation	References a credit entity defined elsewhere in the document.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="LegalEntityReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="LegalEntity"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MainPublication

[Table of contents]

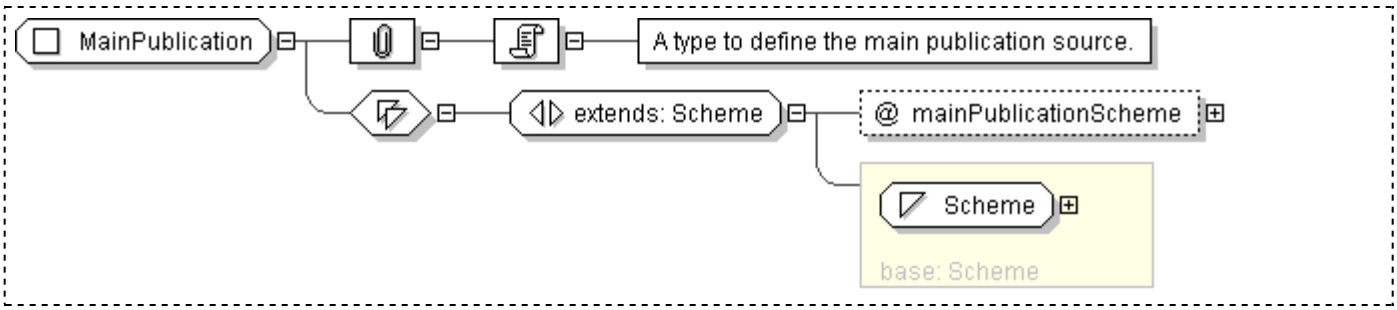
Super-types:	xsd:normalizedString < Scheme (by restriction) < MainPublication (by extension)
Sub-types:	None

Name	MainPublication
Abstract	no
Documentation	A type to define the main publication source.

XML Instance Representation

```
<...
 mainPublicationScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MainPublication">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="mainPublicationScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/inflation-main-publication"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ManualExercise

[Table of contents]

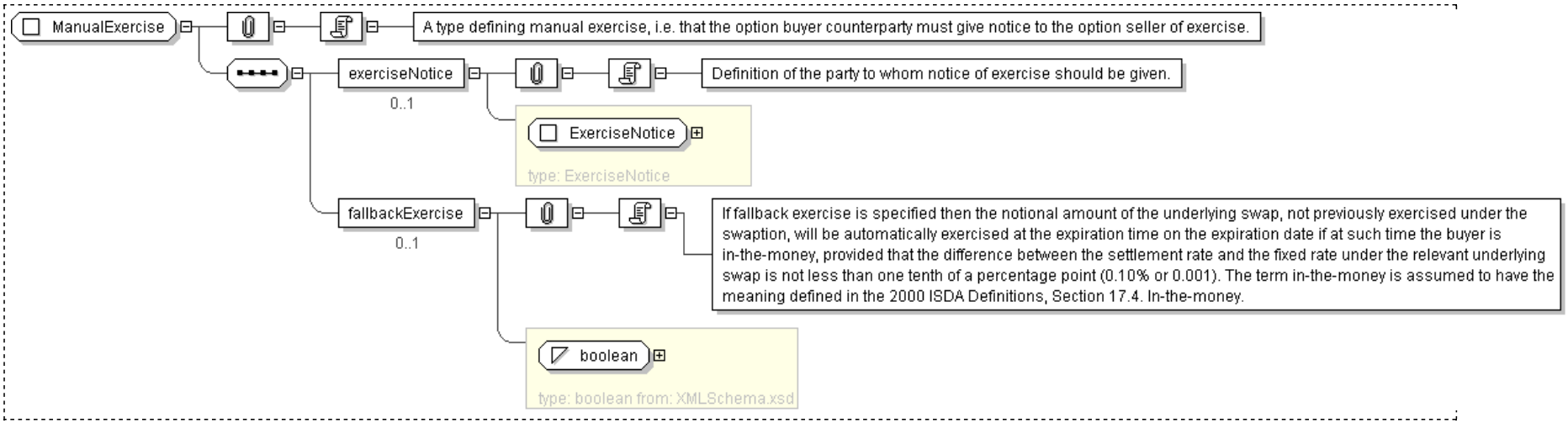
Super-types:	None
Sub-types:	None

Name	ManualExercise
Used by (from the same schema document)	Complex Type ExerciseProcedure
Abstract	no
Documentation	A type defining manual exercise, i.e. that the option buyer counterparty must give notice to the option seller of exercise.

XML Instance Representation

```
<...>
  <exerciseNotice> ExerciseNotice </exerciseNotice> [0..1]
  'Definition of the party to whom notice of exercise should be given.'

  <fallbackExercise> xsd:boolean </fallbackExercise> [0..1]
  'If fallback exercise is specified then the notional amount of the underlying swap, not previously exercised under the swaption, will
  be automatically exercised at the expiration time on the expiration date if at such time the buyer is in-the-money, provided that the
  difference between the settlement rate and the fixed rate under the relevant underlying swap is not less than one tenth of a
  percentage point (0.10% or 0.001). The term in-the-money is assumed to have the meaning defined in the 2000 ISDA Definitions, Section
  17.4. In-the-money.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ManualExercise">
  <xsd:sequence>
    <xsd:element name="exerciseNotice" type=" ExerciseNotice " minOccurs="0"/>
    <xsd:element name="fallbackExercise" type=" xsd:boolean " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterAgreement

[Table of contents]

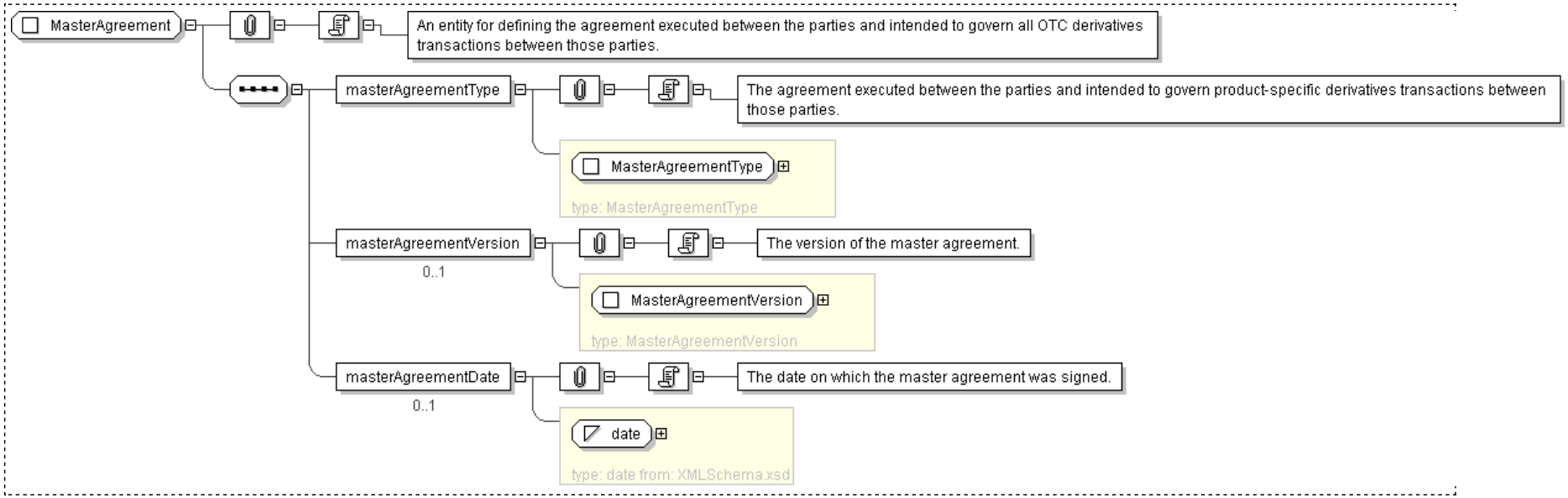
Super-types:	None
Sub-types:	None

Name	MasterAgreement
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	An entity for defining the agreement executed between the parties and intended to govern all OTC derivatives transactions between those parties.

XML Instance Representation

```
<...>
  <masterAgreementType> MasterAgreementType </masterAgreementType> [1]
  'The agreement executed between the parties and intended to govern product-specific derivatives transactions between those parties.'

  <masterAgreementVersion> MasterAgreementVersion </masterAgreementVersion> [0..1]
  'The version of the master agreement.'

  <masterAgreementDate> xsd:date </masterAgreementDate> [0..1]
  'The date on which the master agreement was signed.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterAgreement">
  <xsd:sequence>
    <xsd:element name="masterAgreementType" type=" MasterAgreementType "/>
    <xsd:element name="masterAgreementVersion" type=" MasterAgreementVersion " minOccurs="0"/>
```

```
    <xsd:element name="masterAgreementDate" type=" xsd:date " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterAgreementType

[Table of contents]

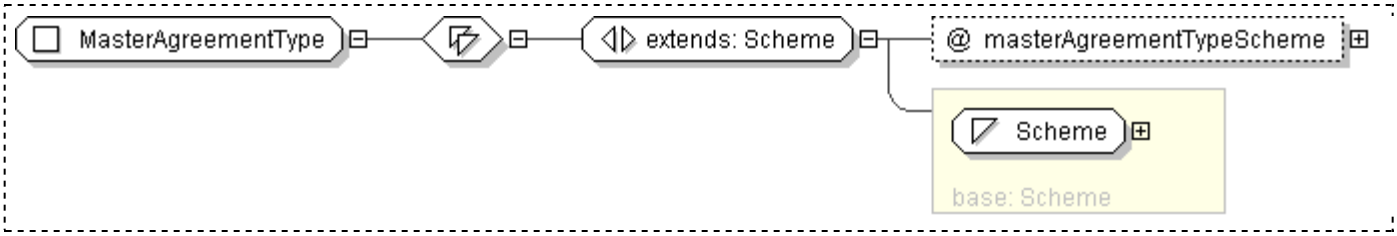
Super-types:	xsd:normalizedString < Scheme (by restriction) < MasterAgreementType (by extension)
Sub-types:	None

Name	MasterAgreementType
Used by (from the same schema document)	Complex Type MasterAgreement
Abstract	no

XML Instance Representation

```
<...
 masterAgreementTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterAgreementType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="masterAgreementTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/master-agreement-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterAgreementVersion

[Table of contents]

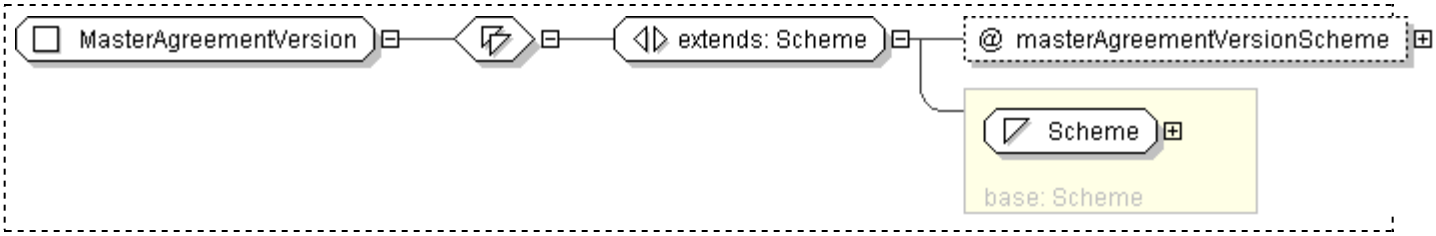
Super-types:	xsd:normalizedString < Scheme (by restriction) < MasterAgreementVersion (by extension)
Sub-types:	None

Name	MasterAgreementVersion
Used by (from the same schema document)	Complex Type MasterAgreement
Abstract	no

XML Instance Representation

```
<...
 masterAgreementVersionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterAgreementVersion">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="masterAgreementVersionScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/master-agreement-version"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterConfirmation

[Table of contents]

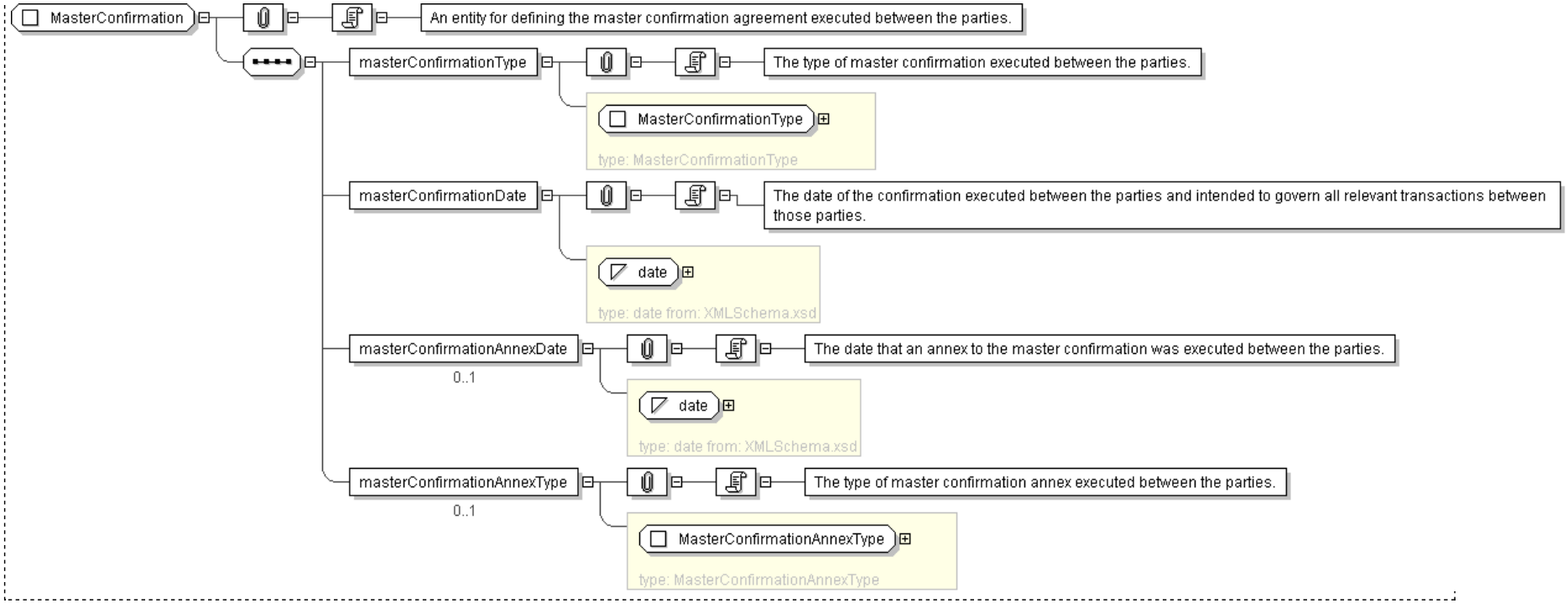
Super-types:	None
Sub-types:	None

Name	MasterConfirmation
Used by (from the same schema document)	Complex Type Documentation
Abstract	no
Documentation	An entity for defining the master confirmation agreement executed between the parties.

XML Instance Representation

```
<...>
   <masterConfirmationType> MasterConfirmationType </masterConfirmationType> [1]
   'The type of master confirmation executed between the parties.'

   <masterConfirmationDate> xsd:date </masterConfirmationDate> [1]
   'The date of the confirmation executed between the parties and intended to govern all relevant transactions between those parties.'

   <masterConfirmationAnnexDate> xsd:date </masterConfirmationAnnexDate> [0..1]
   'The date that an annex to the master confirmation was executed between the parties.'

   <masterConfirmationAnnexType> MasterConfirmationAnnexType </masterConfirmationAnnexType> [0..1]
   'The type of master confirmation annex executed between the parties.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterConfirmation">
  <xsd:sequence>
    <xsd:element name="masterConfirmationType" type=" MasterConfirmationType "/>
    <xsd:element name="masterConfirmationDate" type=" xsd:date "/>
    <xsd:element name="masterConfirmationAnnexDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="masterConfirmationAnnexType" type=" MasterConfirmationAnnexType " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterConfirmationAnnexType

[Table of contents]

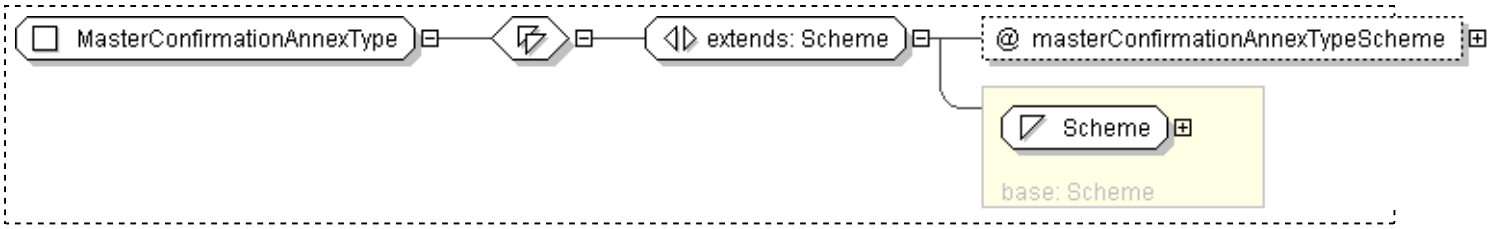
Super-types:	xsd:normalizedString < Scheme (by restriction) < MasterConfirmationAnnexType (by extension)
Sub-types:	None

Name	MasterConfirmationAnnexType
Used by (from the same schema document)	Complex Type MasterConfirmation
Abstract	no

XML Instance Representation

```
<...
 masterConfirmationAnnexTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterConfirmationAnnexType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="masterConfirmationAnnexTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/master-confirmation-annex-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MasterConfirmationType

[Table of contents]

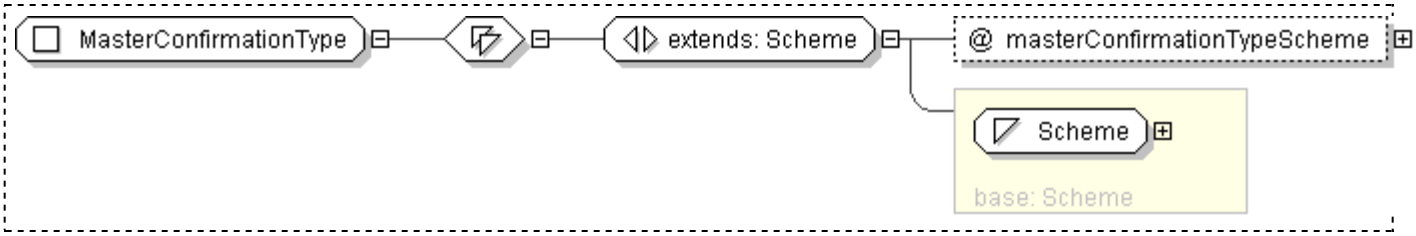
Super-types:	xsd:normalizedString < Scheme (by restriction) < MasterConfirmationType (by extension)
Sub-types:	None

Name	MasterConfirmationType
Used by (from the same schema document)	Complex Type MasterConfirmation
Abstract	no

XML Instance Representation

```
<...
 masterConfirmationTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MasterConfirmationType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="masterConfirmationTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/master-confirmation-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Math

[Table of contents]

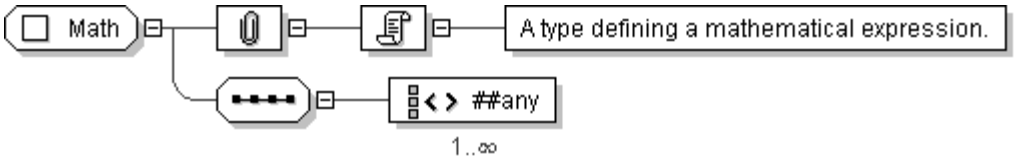
Super-types:	None
Sub-types:	None

Name	Math
Used by (from the same schema document)	Complex Type Formula
Abstract	no
Documentation	A type defining a mathematical expression.

XML Instance Representation

```
<...>
   <!-- Mixed content -->
   Allow any elements from any namespace (skip validation). [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Math" mixed="true">
   <xsd:sequence>
      <xsd:any namespace="##any" processContents="skip" maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MatrixTerm

[Table of contents]

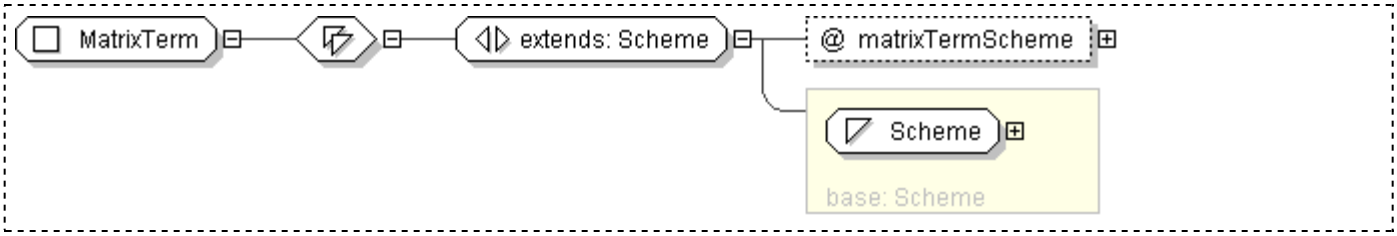
Super-types:	xsd:normalizedString < Scheme (by restriction) < MatrixTerm (by extension)
Sub-types:	None

Name	MatrixTerm
Used by (from the same schema document)	Complex Type ContractualMatrix
Abstract	no

XML Instance Representation

```
<...
 matrixTermScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MatrixTerm">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="matrixTermScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/credit-matrix-transaction-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MatrixType

[Table of contents]

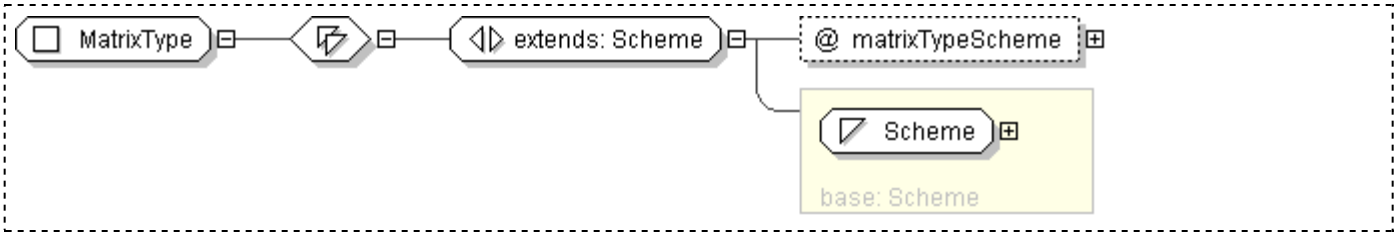
Super-types:	xsd:normalizedString < Scheme (by restriction) < MatrixType (by extension)
Sub-types:	None

Name	MatrixType
Used by (from the same schema document)	Complex Type ContractualMatrix
Abstract	no

XML Instance Representation

```
<...
 matrixTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MatrixType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="matrixTypeScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/matrix-type"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MimeType

[Table of contents]

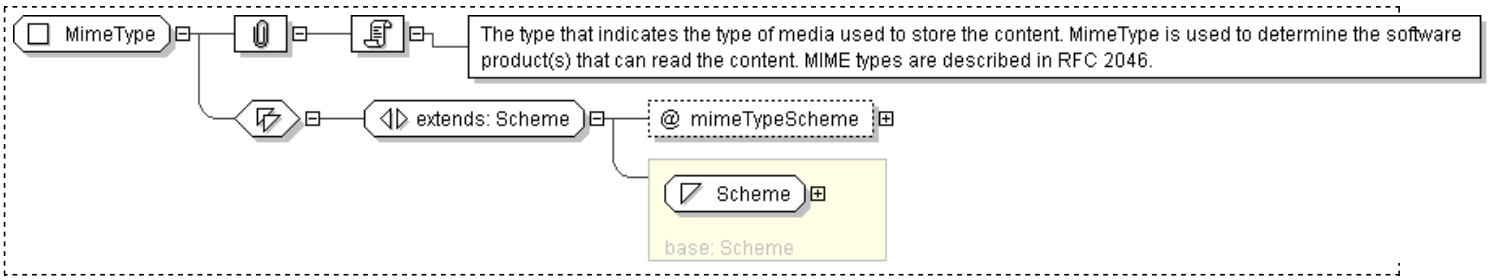
Super-types:	xsd:normalizedString < Scheme (by restriction) < MimeType (by extension)
Sub-types:	None

Name	MimeType
Abstract	no
Documentation	The type that indicates the type of media used to store the content. MimeType is used to determine the software product(s) that can read the content. MIME types are described in RFC 2046.

XML Instance Representation

```
<...
 mimeTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MimeType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="mimeTypeScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Money

[Table of contents]

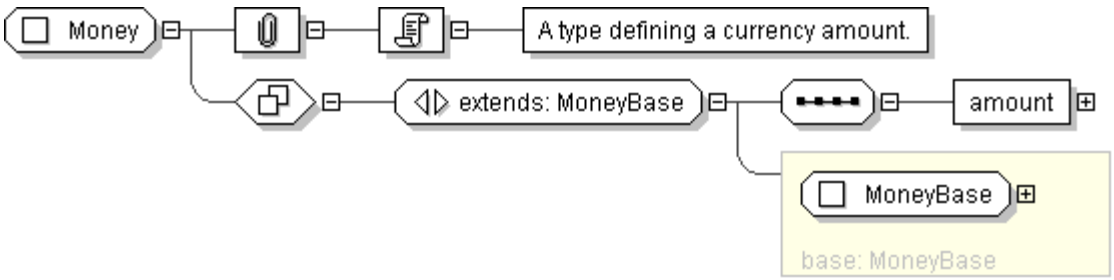
Super-types:	MoneyBase < Money (by extension)
Sub-types:	• NotionalAmount (by extension)

Name	Money
Used by (from the same schema document)	Complex Type Payment , Complex Type Payment , Complex Type SimplePayment , Complex Type SplitSettlement , Complex Type StubValue , Model Group PaymentDiscounting.model , Model Group Premium.model , Model Group SettlementAmountOrCurrency.model
Abstract	no
Documentation	A type defining a currency amount.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

   <amount> xsd:decimal </amount> [1]
   'The monetary quantity in currency units.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Money">
   <xsd:complexContent>
      <xsd:extension base=" MoneyBase ">
         <xsd:sequence>
            <xsd:element name="amount" type=" xsd:decimal "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MoneyBase

[Table of contents]

Super-types: None

Sub-types:

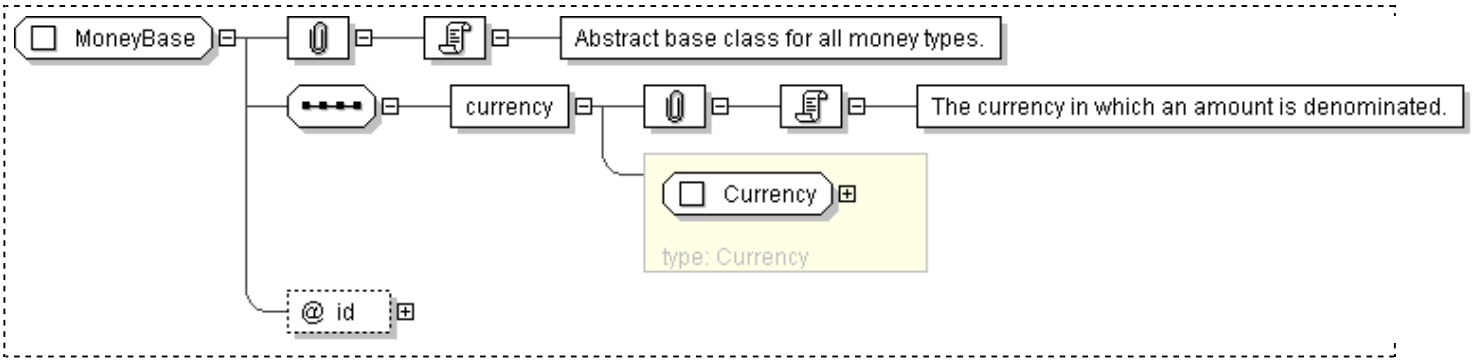
- Money (by extension)
  - NotionalAmount (by extension)
- NonNegativeMoney (by extension)
  - FutureValueAmount (by extension)
- PositiveMoney (by extension)

Name	MoneyBase
Abstract	yes
Documentation	Abstract base class for all money types.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MoneyBase" abstract="true">
   <xsd:sequence>
      <xsd:element name="currency" type=" Currency "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: MultipleExercise

[Table of contents]

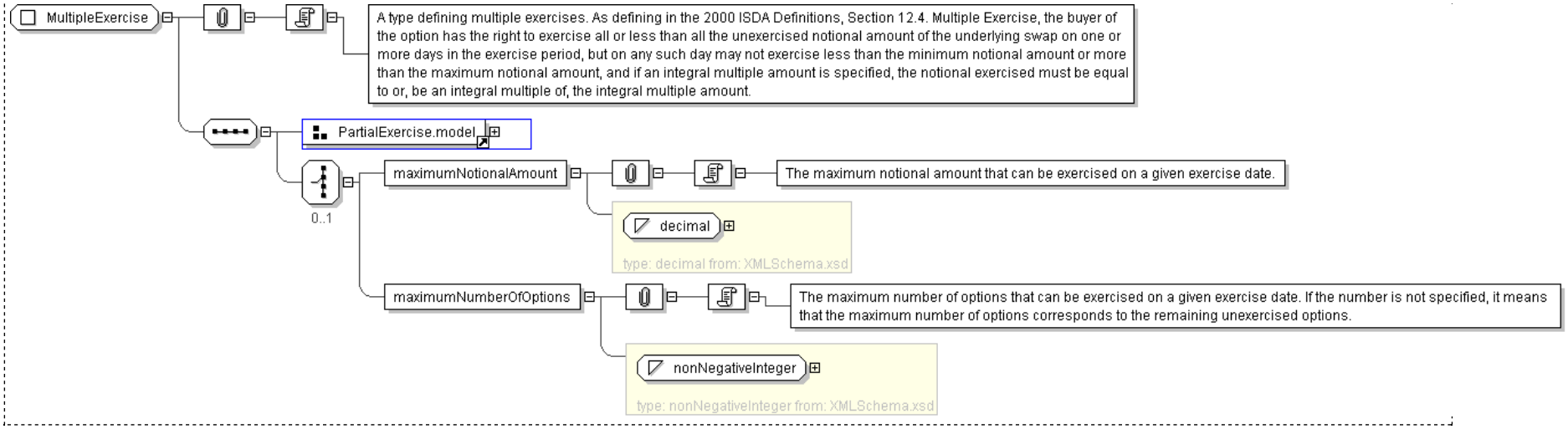
Super-types:	None
Sub-types:	None

Name	MultipleExercise
Used by (from the same schema document)	Complex Type AmericanExercise , Complex Type BermudaExercise
Abstract	no
Documentation	A type defining multiple exercises. As defining in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on any such day may not exercise less than the minimum notional amount or more than the maximum notional amount, and if an integral multiple amount is specified, the notional exercised must be equal to or, be an integral multiple of, the integral multiple amount.

XML Instance Representation

```
<...>
   <notionalReference> NotionalReference </notionalReference> [0..*]
   'A pointer style reference to the associated notional schedule defined elsewhere in the document. This element has been made optional as part of its
   integration in the OptionBaseExtended, because not required for the options on securities.'

   <integralMultipleAmount> xsd:decimal </integralMultipleAmount> [0..1]
   'A notional amount which restricts the amount of notional that can be exercised when partial exercise or multiple exercise is applicable. The integral
   multiple amount defines a lower limit of notional that can be exercised and also defines a unit multiple of notional that can be exercised, i.e. only
   integer multiples of this amount can be exercised.'

   Start Choice [1]
      <minimumNotionalAmount> xsd:decimal </minimumNotionalAmount> [1]
      'The minimum notional amount that can be exercised on a given exercise date. See multipleExercise.'

      <minimumNumberOfOptions> xsd:nonNegativeInteger </minimumNumberOfOptions> [1]
      'The minimum number of options that can be exercised on a given exercise date.'

   End Choice
   Start Choice [0..1]
      <maximumNotionalAmount> xsd:decimal </maximumNotionalAmount> [1]
      'The maximum notional amount that can be exercised on a given exercise date.'

      <maximumNumberOfOptions> xsd:nonNegativeInteger </maximumNumberOfOptions> [1]
      'The maximum number of options that can be exercised on a given exercise date. If the number is not specified, it means that the maximum number of
      options corresponds to the remaining unexercised options.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MultipleExercise">
  <xsd:sequence>
    <xsd:group ref=" PartialExercise.model "/>
    <xsd:choice minOccurs="0">
      <xsd:element name="maximumNotionalAmount" type=" xsd:decimal "/>
      <xsd:element name="maximumNumberOfOptions" type=" xsd:nonNegativeInteger "/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonNegativeAmountSchedule

[Table of contents]

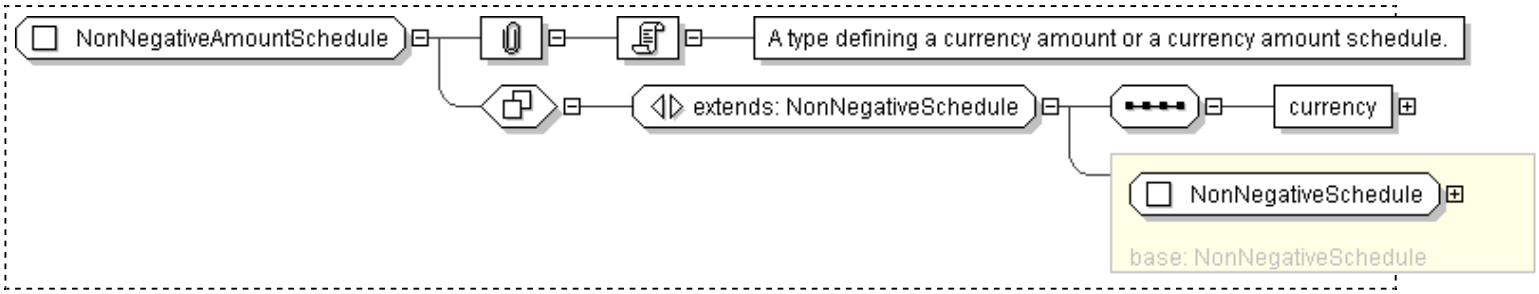
Super-types:	NonNegativeSchedule < NonNegativeAmountSchedule (by extension)
Sub-types:	None

Name	NonNegativeAmountSchedule
Abstract	no
Documentation	A type defining a currency amount or a currency amount schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> NonNegativeDecimal </initialValue> [1]
   'The non-negative initial rate or amount, as the case may be. An initial rate of 5%
   would be represented as 0.05.'

   <step> NonNegativeStep </step> [0..*]
   'The schedule of step date and non-negative value pairs. On each step date the
   associated step value becomes effective. A list of steps may be ordered in the
   document by ascending step date. An FpML document containing an unordered list of
   steps is still regarded as a conformant document.'

   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonNegativeAmountSchedule">
   <xsd:complexContent>
      <xsd:extension base=" NonNegativeSchedule ">
         <xsd:sequence>
            <xsd:element name="currency" type=" Currency "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

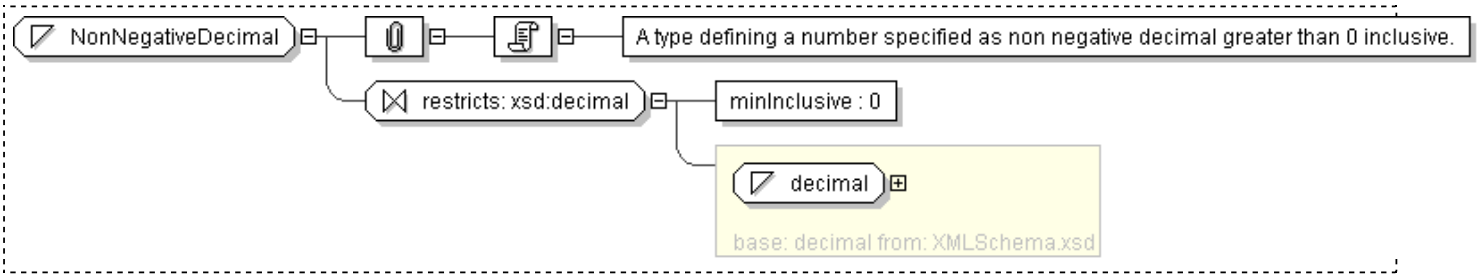
Simple Type: NonNegativeDecimal

[Table of contents]

Super-types:	xsd:decimal < NonNegativeDecimal (by restriction)
Sub-types:	None

Name	NonNegativeDecimal
Used by (from the same schema document)	Complex Type NonNegativeMoney , Complex Type NonNegativeSchedule , Complex Type NonNegativeStep
Content	• Base XSD Type: decimal • *value* >= 0
Documentation	A type defining a number specified as non negative decimal greater than 0 inclusive.

Diagram



Schema Component Representation

```
<xsd:simpleType name="NonNegativeDecimal">
  <xsd:restriction base=" xsd:decimal ">
    <xsd:minInclusive value="0"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonNegativeMoney

[Table of contents]

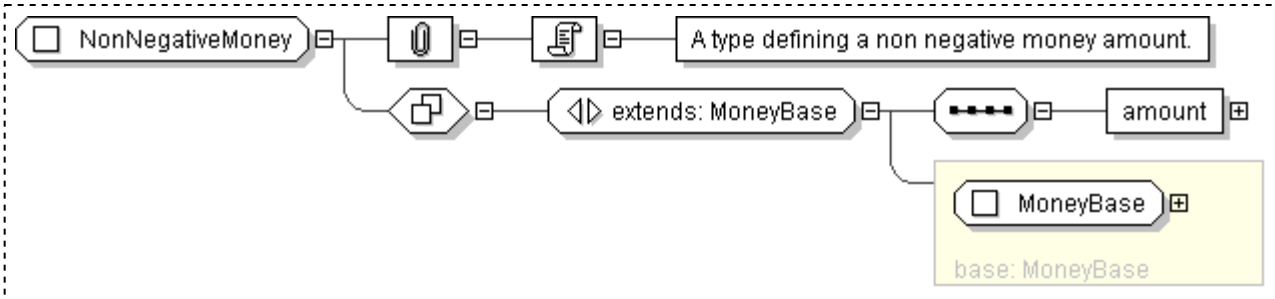
Super-types:	MoneyBase < NonNegativeMoney (by extension)
Sub-types:	• FutureValueAmount (by extension)

Name	NonNegativeMoney
Used by (from the same schema document)	Complex Type NonNegativePayment
Abstract	no
Documentation	A type defining a non negative money amount.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

   <amount> NonNegativeDecimal </amount> [1]
    'The non negative monetary quantity in currency units.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonNegativeMoney">
   <xsd:complexContent>
      <xsd:extension base=" MoneyBase ">
         <xsd:sequence>
            <xsd:element name="amount" type=" NonNegativeDecimal "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonNegativePayment

[Table of contents]

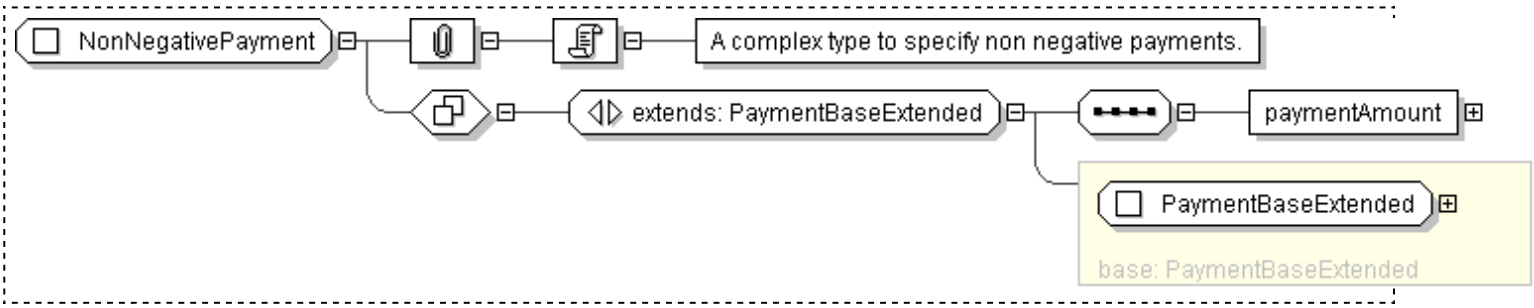
Super-types:	PaymentBase < PaymentBaseExtended (by extension) < NonNegativePayment (by extension)
Sub-types:	None

Name	NonNegativePayment
Abstract	no
Documentation	A complex type to specify non negative payments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this
    structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this
    structure.'

   <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
    'The payment date, which can be expressed as either an adjustable or relative
    date.'

   <paymentAmount> NonNegativeMoney </paymentAmount> [1]
    'Non negative payment amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonNegativePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBaseExtended ">
         <xsd:sequence>
            <xsd:element name="paymentAmount" type=" NonNegativeMoney "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonNegativeSchedule

[Table of contents]

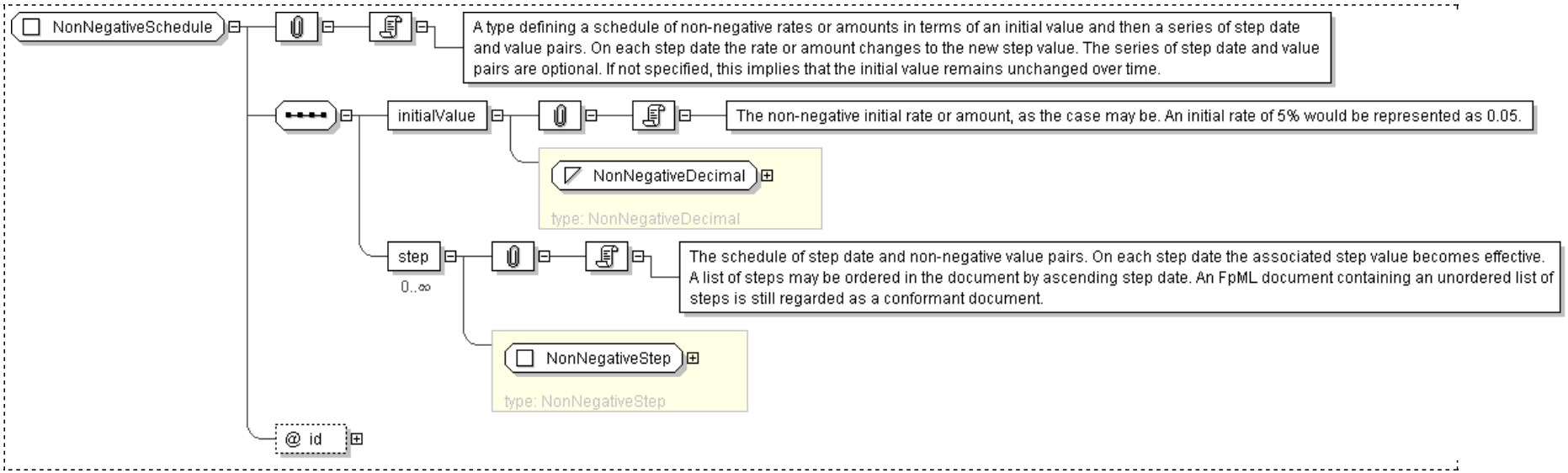
Super-types:	None
Sub-types:	• NonNegativeAmountSchedule (by extension)

Name	NonNegativeSchedule
Abstract	no
Documentation	A type defining a schedule of non-negative rates or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> NonNegativeDecimal </initialValue> [1]
    'The non-negative initial rate or amount, as the case may be. An initial rate of 5% would be represented as 0.05.'

   <step> NonNegativeStep </step> [0..*]
    'The schedule of step date and non-negative value pairs. On each step date the associated step value becomes effective. A list of
    steps may be ordered in the document by ascending step date. An FpML document containing an unordered list of steps is still
    regarded as a conformant document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonNegativeSchedule">
  <xsd:sequence>
    <xsd:element name="initialValue" type=" NonNegativeDecimal "/>
```

```
    <xsd:element name="step" type=" NonNegativeStep " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NonNegativeStep

[Table of contents]

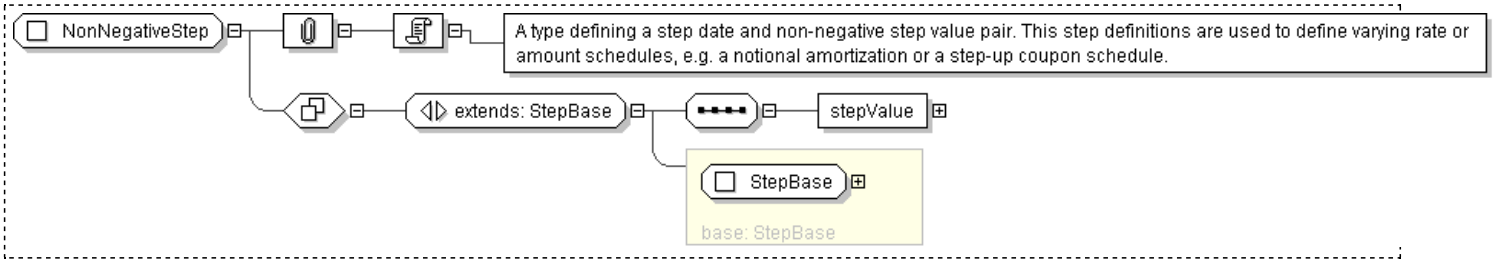
Super-types:	StepBase < NonNegativeStep (by extension)
Sub-types:	None

Name	NonNegativeStep
Used by (from the same schema document)	Complex Type NonNegativeSchedule
Abstract	no
Documentation	A type defining a step date and non-negative step value pair. This step definitions are used to define varying rate or amount schedules, e.g. a notional amortization or a step-up coupon schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <stepDate> xsd:date </stepDate> [1]
   'The date on which the associated stepValue becomes effective. This day may be subject to adjustment in
   accordance with a business day convention.'

   <stepValue> NonNegativeDecimal </stepValue> [1]
   'The non-negative rate or amount which becomes effective on the associated stepDate. A rate of 5% would be
   represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NonNegativeStep">
  <xsd:complexContent>
    <xsd:extension base=" StepBase ">
      <xsd:sequence>
        <xsd:element name="stepValue" type=" NonNegativeDecimal "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotionalAmount

[Table of contents]

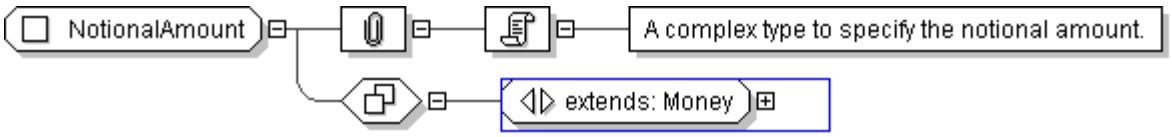
Super-types:	MoneyBase < Money (by extension) < NotionalAmount (by extension)
Sub-types:	None

Name	NotionalAmount
Abstract	no
Documentation	A complex type to specify the notional amount.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

   <amount> xsd:decimal </amount> [1]
    'The monetary quantity in currency units.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotionalAmount">
   <xsd:complexContent>
      <xsd:extension base=" Money "/>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotionalAmountReference

[Table of contents]

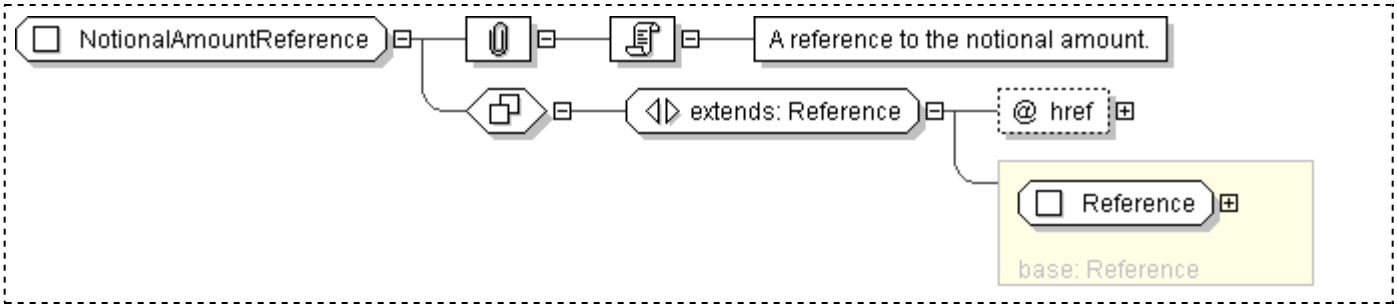
Super-types:	Reference < NotionalAmountReference (by extension)
Sub-types:	None

Name	NotionalAmountReference
Abstract	no
Documentation	A reference to the notional amount.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotionalAmountReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: NotionalReference

[Table of contents]

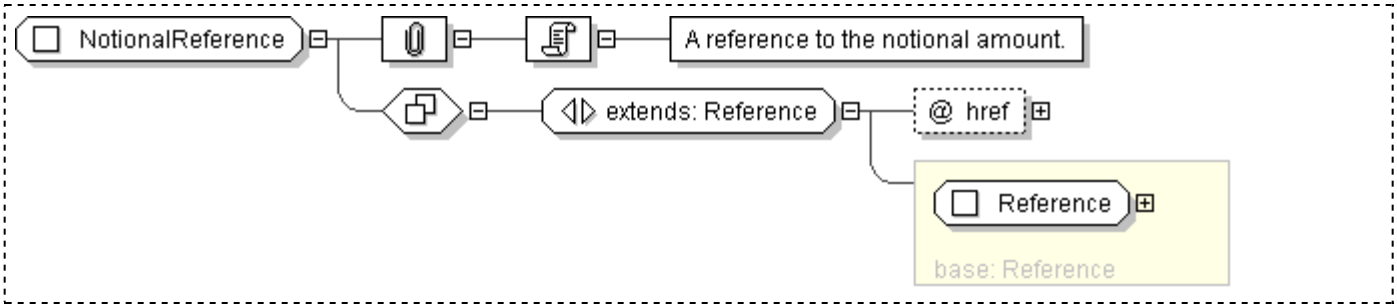
Super-types:	Reference < NotionalReference (by extension)
Sub-types:	None

Name	NotionalReference
Used by (from the same schema document)	Complex Type ExerciseFee , Model Group PartialExercise.model
Abstract	no
Documentation	A reference to the notional amount.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NotionalReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Notional"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Offset

[Table of contents]

Super-types: Period < Offset (by extension)

Sub-types:

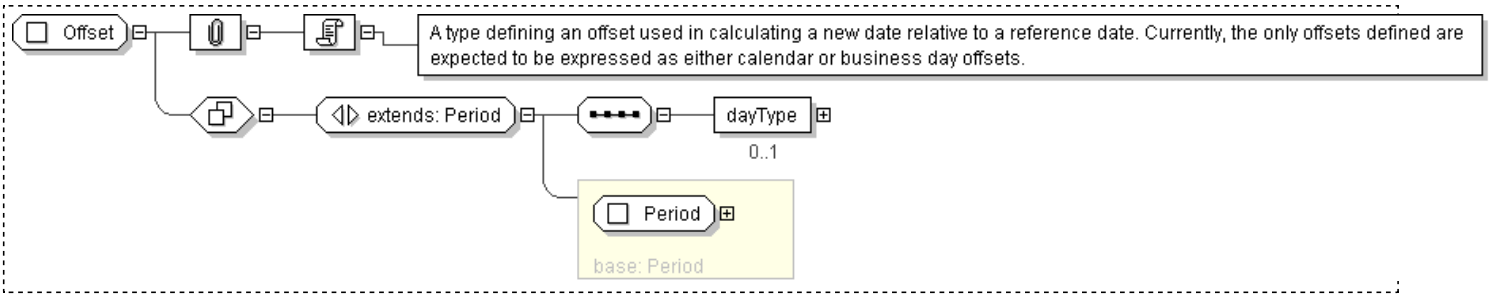
- DateOffset (by extension)
- RelativeDateOffset (by extension)
  - AdjustedRelativeDateOffset (by extension)
  - RelativeDates (by extension)

Name	Offset
Used by (from the same schema document)	Complex Type OffsetPrevailingTime
Abstract	no
Documentation	A type defining an offset used in calculating a new date relative to a reference date. Currently, the only offsets defined are expected to be expressed as either calendar or business day offsets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
    'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset
    relative to another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
    'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0
    (zero) then period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
    'In the case of an offset specified as a number of days, this element defines whether consideration is
    given as to whether a day is a good business day or not. If a day type of business days is specified
    then non-business days are ignored when calculating the offset. The financial business centers to use
    for determination of business days are implied by the context in which this element is used. This
    element must only be included when the offset is specified as a number of days. If the offset is zero
    days then the dayType element should not be included.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Offset">
  <xsd:complexContent>
    <xsd:extension base=" Period ">
      <xsd:sequence>
        <xsd:element name="dayType" type=" DayTypeEnum " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: OffsetPrevailingTime

[Table of contents]

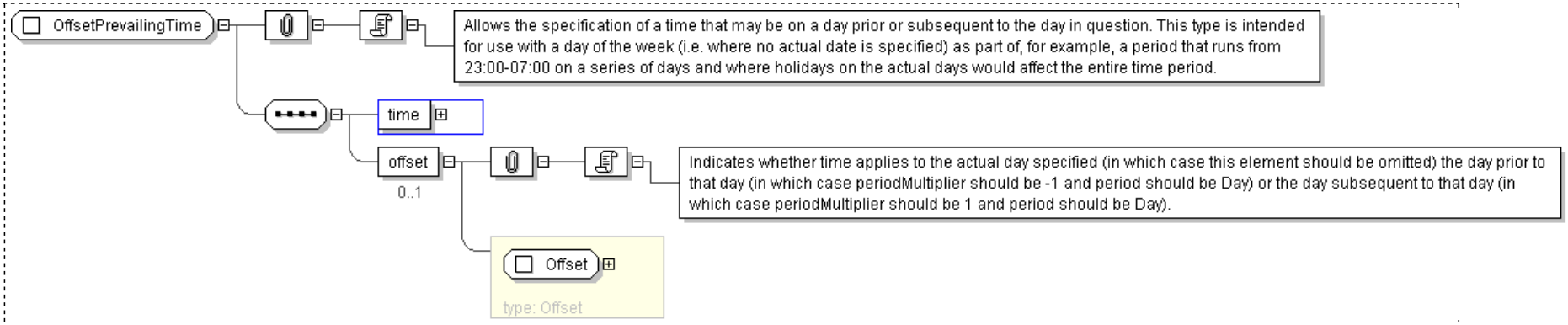
Super-types:	None
Sub-types:	None

Name	OffsetPrevailingTime
Abstract	no
Documentation	Allows the specification of a time that may be on a day prior or subsequent to the day in question. This type is intended for use with a day of the week (i.e. where no actual date is specified) as part of, for example, a period that runs from 23:00-07:00 on a series of days and where holidays on the actual days would affect the entire time period.

XML Instance Representation

```
<...>
   <time> PrevailingTime </time> [1]
   <offset> Offset </offset> [0..1]
   'Indicates whether time applies to the actual day specified (in which case this element should be omitted) the day prior to that day
   (in which case periodMultiplier should be -1 and period should be Day) or the day subsequent to that day (in which case
   periodMultiplier should be 1 and period should be Day).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="OffsetPrevailingTime">
   <xsd:sequence>
      <xsd:element name="time" type=" PrevailingTime "/>
      <xsd:element name="offset" type=" Offset " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartialExercise

[Table of contents]

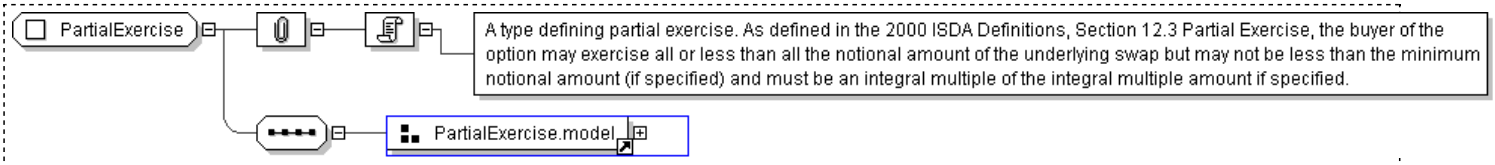
Super-types:	None
Sub-types:	None

Name	PartialExercise
Used by (from the same schema document)	Complex Type EuropeanExercise
Abstract	no
Documentation	A type defining partial exercise. As defined in the 2000 ISDA Definitions, Section 12.3 Partial Exercise, the buyer of the option may exercise all or less than all the notional amount of the underlying swap but may not be less than the minimum notional amount (if specified) and must be an integral multiple of the integral multiple amount if specified.

XML Instance Representation

```
<...>
  <notionalReference> NotionalReference </notionalReference> [0..*]
  'A pointer style reference to the associated notional schedule defined elsewhere in the document. This
  element has been made optional as part of its integration in the OptionBaseExtended, because not required for
  the options on securities.'

  <integralMultipleAmount> xsd:decimal </integralMultipleAmount> [0..1]
  'A notional amount which restricts the amount of notional that can be exercised when partial exercise or
  multiple exercise is applicable. The integral multiple amount defines a lower limit of notional that can be
  exercised and also defines a unit multiple of notional that can be exercised, i.e. only integer multiples of
  this amount can be exercised.'

  Start Choice [1]
     <minimumNotionalAmount> xsd:decimal </minimumNotionalAmount> [1]
     'The minimum notional amount that can be exercised on a given exercise date. See multipleExercise.'

     <minimumNumberOfOptions> xsd:nonNegativeInteger </minimumNumberOfOptions> [1]
     'The minimum number of options that can be exercised on a given exercise date.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartialExercise">
  <xsd:sequence>
    <xsd:group ref=" PartialExercise.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Party

[Table of contents]

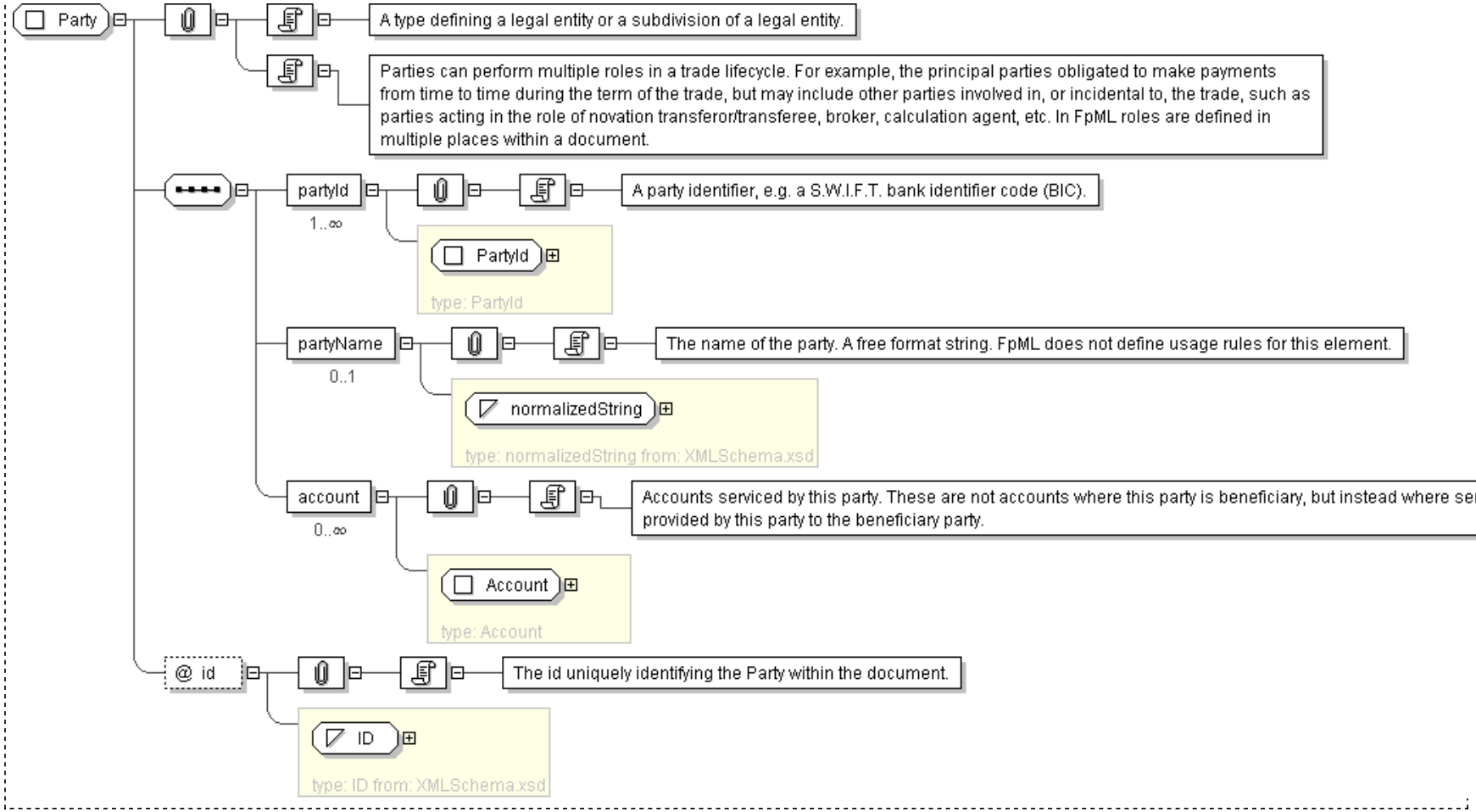
Super-types:	None
Sub-types:	None

Name	Party
Abstract	no
Documentation	A type defining a legal entity or a subdivision of a legal entity. Parties can perform multiple roles in a trade lifecycle. For example, the principal parties obligated to make payments from time to time during the term of the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in multiple places within a document.

XML Instance Representation

```
<...
 id=" xsd:ID [1]
 'The id uniquely identifying the Party within the document.'

 ">
   <partyId> PartyId </partyId> [1..*]
    'A party identifier, e.g. a S.W.I.F.T. bank identifier code (BIC).'

   <partyName> xsd:normalizedString </partyName> [0..1]
    'The name of the party. A free format string. FpML does not define usage rules for this element.'

   <account> Account </account> [0..*]
    'Accounts serviced by this party. These are not accounts where this party is beneficiary, but instead where services are
    provided by this party to the beneficiary party.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Party">
  <xsd:sequence>
    <xsd:element name="partyId" type=" PartyId " maxOccurs="unbounded"/>
    <xsd:element name="partyName" type=" xsd:normalizedString " minOccurs="0"/>
    <xsd:element name="account" type=" Account " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyId

[Table of contents]

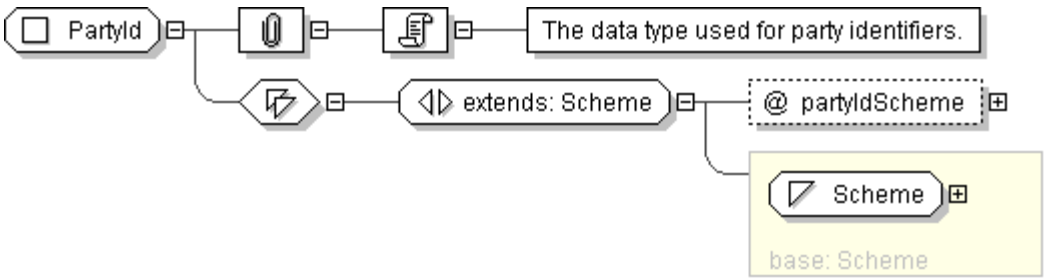
Super-types:	xsd:normalizedString < Scheme (by restriction) < PartyId (by extension)
Sub-types:	None

Name	PartyId
Used by (from the same schema document)	Complex Type Party
Abstract	no
Documentation	The data type used for party identifiers.

XML Instance Representation

```
<...
 partyIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="partyIdScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/ext/iso9362"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyOrAccountReference

[Table of contents]

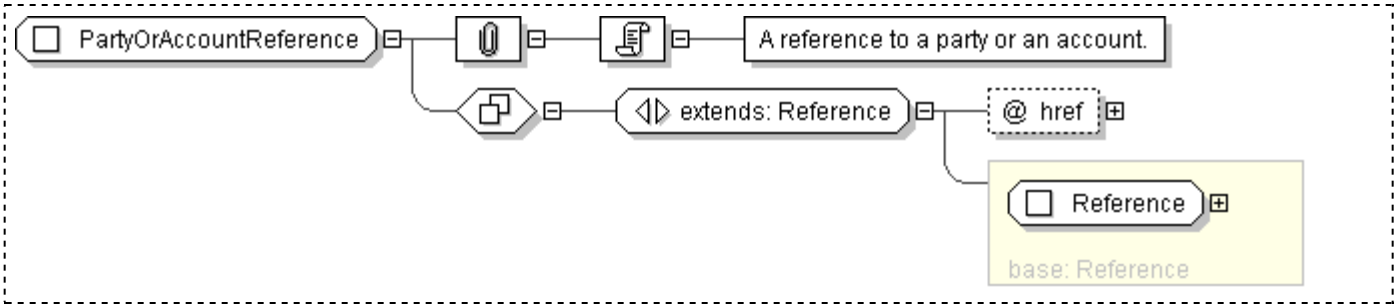
Super-types:	Reference < PartyOrAccountReference (by extension)
Sub-types:	None

Name	PartyOrAccountReference
Used by (from the same schema document)	Model Group PayerReceiver.model , Model Group PayerReceiver.model
Abstract	no
Documentation	A reference to a party or an account.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyOrAccountReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyOrTradeSideReference

[Table of contents]

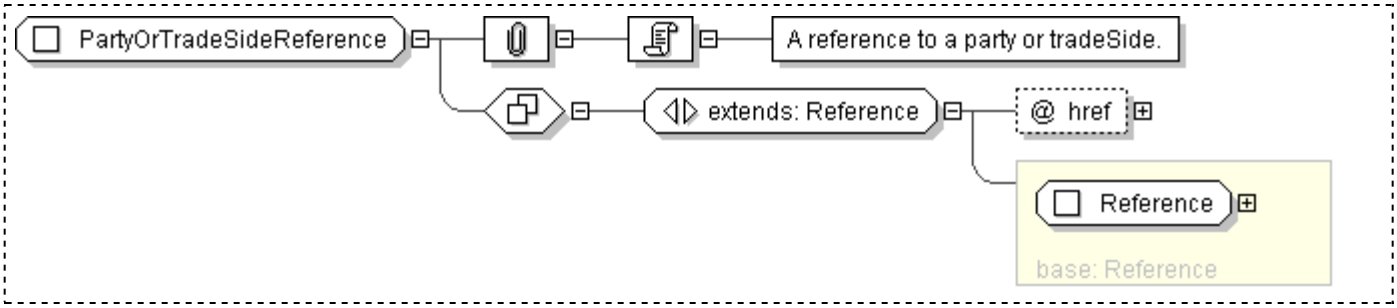
Super-types:	Reference < PartyOrTradeSideReference (by extension)
Sub-types:	None

Name	PartyOrTradeSideReference
Used by (from the same schema document)	Model Group BuyerSeller.model , Model Group BuyerSeller.model
Abstract	no
Documentation	A reference to a party or tradeSide.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyOrTradeSideReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PartyReference

[Table of contents]

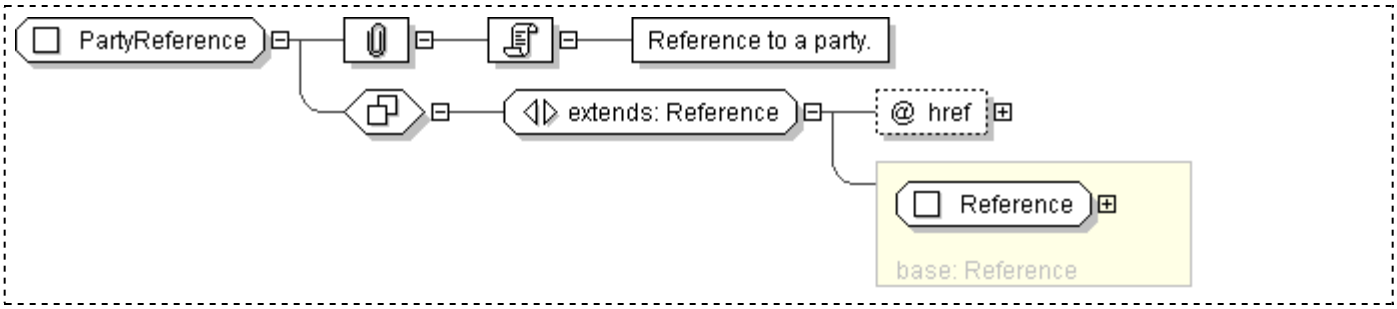
Super-types:	Reference < PartyReference (by extension)
Sub-types:	None

Name	PartyReference
Used by (from the same schema document)	Complex Type Account , Complex Type Beneficiary , Complex Type CalculationAgent , Complex Type CorrespondentInformation , Complex Type ExerciseNotice , Complex Type ExerciseNotice , Complex Type IntermediaryInformation , Complex Type SettlementInstruction
Abstract	no
Documentation	Reference to a party.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PartyReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="Party"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Payment

[Table of contents]

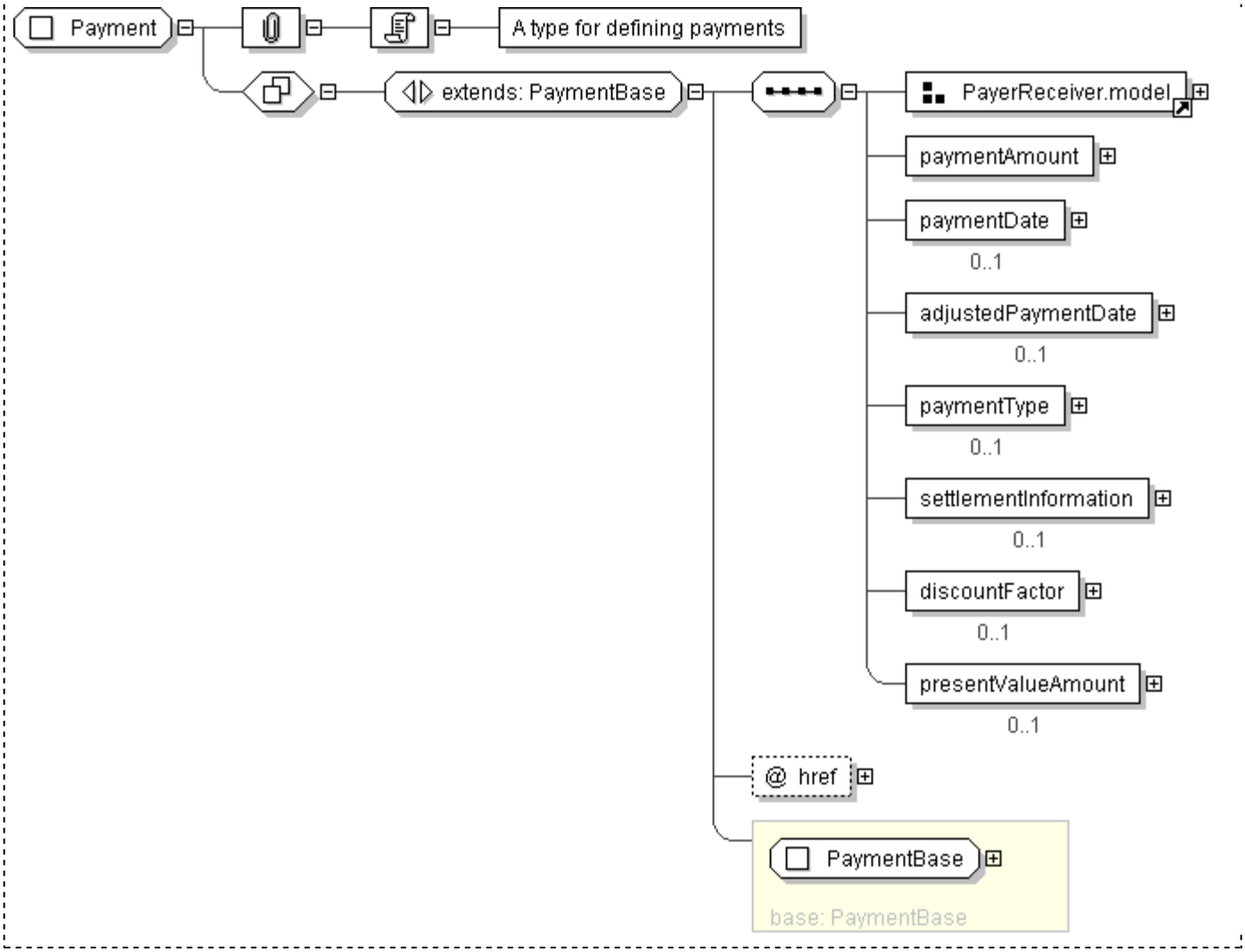
Super-types:	PaymentBase < Payment (by extension)
Sub-types:	None

Name	Payment
Abstract	no
Documentation	A type for defining payments

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 href=" xsd:IDREF [0..1]
'Can be used to reference the yield curve used to estimate the discount factor'

 ">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this
   structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this
   structure.'

   <paymentAmount> Money </paymentAmount> [1]
   'The currency amount of the payment.'

   <paymentDate> AdjustableDate </paymentDate> [0..1]
   'The payment date. This date is subject to adjustment in accordance with any
   applicable business day convention.'

   <adjustedPaymentDate> IdentifiedDate </adjustedPaymentDate> [0..1]
   'The adjusted payment date. This date should already be adjusted for any
   applicable business day convention. This component is not intended for use in
   trade confirmation but my be specified to allow the fee structure to also serve
   as a cashflow type component (all dates the the Cashflows type are adjusted
   payment dates).'

   <paymentType> PaymentType </paymentType> [0..1]
   'A classification of the type of fee or additional payment, e.g. brokerage,
   upfront fee etc. FpML does not define domain values for this element.'

   <settlementInformation> SettlementInformation </settlementInformation> [0..1]
   'The information required to settle a currency payment that results from a
   trade.'

   <discountFactor> xsd:decimal </discountFactor> [0..1]
   'The value representing the discount factor used to calculate the present value
   of the cash flow.'

   <presentValueAmount> Money </presentValueAmount> [0..1]
   'The amount representing the present value of the forecast payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Payment">
  <xsd:complexContent>
    <xsd:extension base=" PaymentBase ">
      <xsd:sequence>
        <xsd:group ref=" PayerReceiver.model "/>
        <xsd:element name="paymentAmount" type=" Money "/>
        <xsd:element name="paymentDate" type=" AdjustableDate " minOccurs="0"/>
        <xsd:element name="adjustedPaymentDate" type=" IdentifiedDate "
        minOccurs="0"/>
        <xsd:element name="paymentType" type=" PaymentType " minOccurs="0"/>
        <xsd:element name="settlementInformation" type=" SettlementInformation "
        minOccurs="0"/>
        <xsd:element name="discountFactor" type=" xsd:decimal " minOccurs="0"/>
        <xsd:element name="presentValueAmount" type=" Money " minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="href" type=" xsd:IDREF " reference="PricingStructure"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentBase

[Table of contents]

Super-types: None

Sub-types:

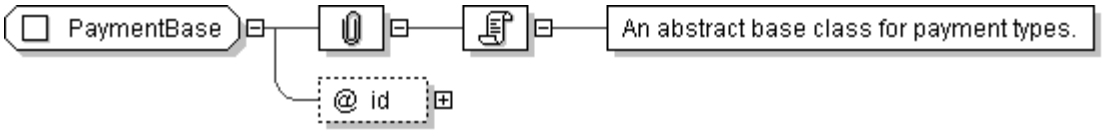
- Payment (by extension)
- PaymentBaseExtended (by extension)
  - NonNegativePayment (by extension)
  - PositivePayment (by extension)
- SimplePayment (by extension)

Name	PaymentBase
Abstract	yes
Documentation	An abstract base class for payment types.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentBase" abstract="true">
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentBaseExtended

[Table of contents]

Super-types: PaymentBase < PaymentBaseExtended (by extension)

Sub-types:

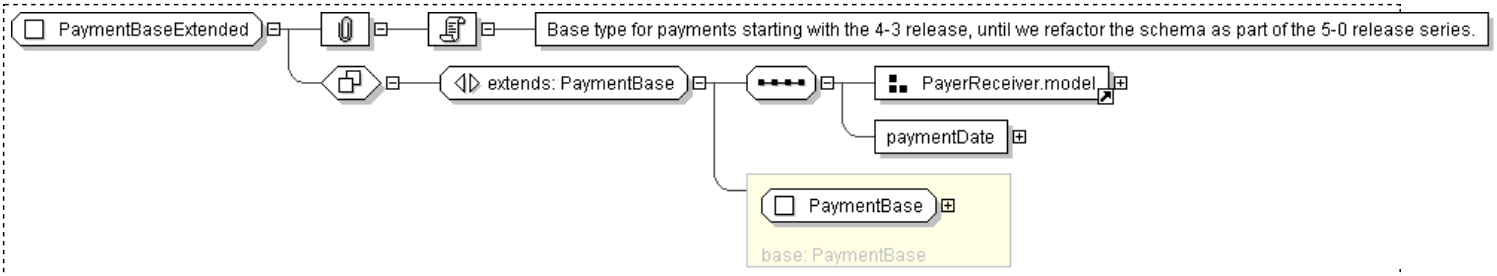
- NonNegativePayment (by extension)
- PositivePayment (by extension)

Name	PaymentBaseExtended
Abstract	yes
Documentation	Base type for payments starting with the 4-3 release, until we refactor the schema as part of the 5-0 release series.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
   'The payment date, which can be expressed as either an adjustable or relative date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentBaseExtended" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="paymentDate" type=" AdjustableOrRelativeDate "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentCurrency

[Table of contents]

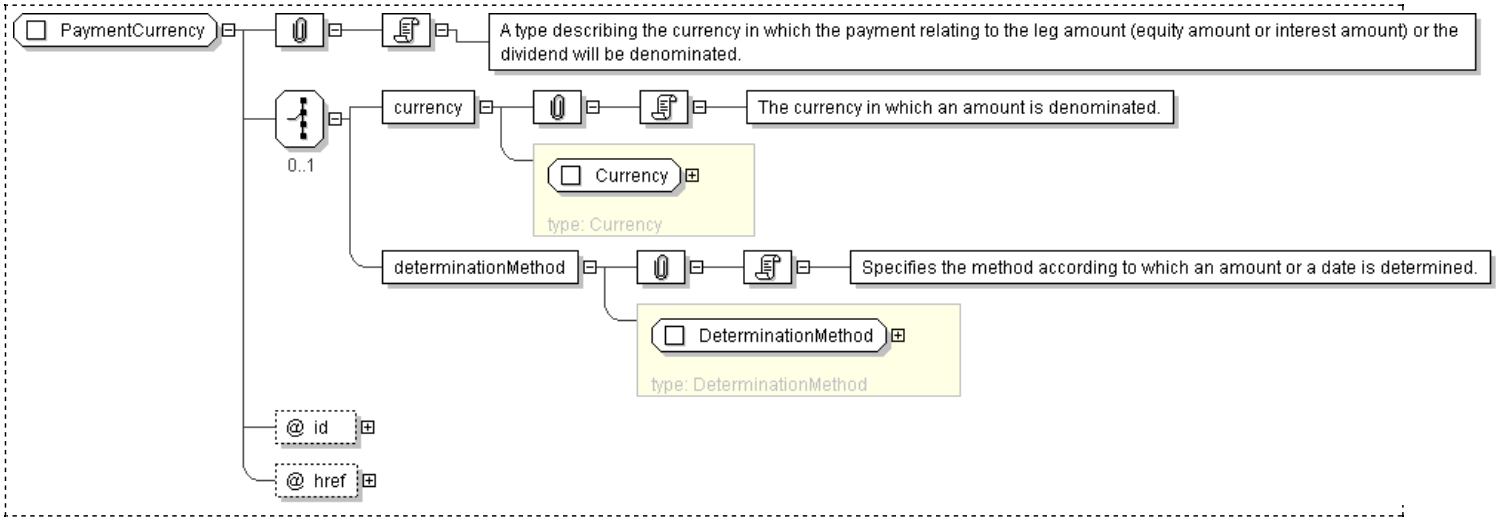
Super-types:	None
Sub-types:	None

Name	PaymentCurrency
Abstract	no
Documentation	A type describing the currency in which the payment relating to the leg amount (equity amount or interest amount) or the dividend will be denominated.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 href=" xsd:IDREF [0..1]">
   Start Choice [0..1]
      <currency> Currency </currency> [1]
      'The currency in which an amount is denominated.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentCurrency" deprecated="true" deprecatedReason="The model is wrong since it has an
intradocument reference that is not clear. Current PaymentCurrency model and elements using this type are deprecated.
Instead, a choice between currency, determinationMethod, and currencyReference (of type CurrencyReference) are
inserted at the same level as the elements using PaymentCurrency.">
  <xsd:choice minOccurs="0">
    <xsd:element name="currency" type=" Currency "/>
    <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
  </xsd:choice>
  <xsd:attribute name="id" type=" xsd:ID "/>
  <xsd:attribute name="href" type=" xsd:IDREF "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentReference

[Table of contents]

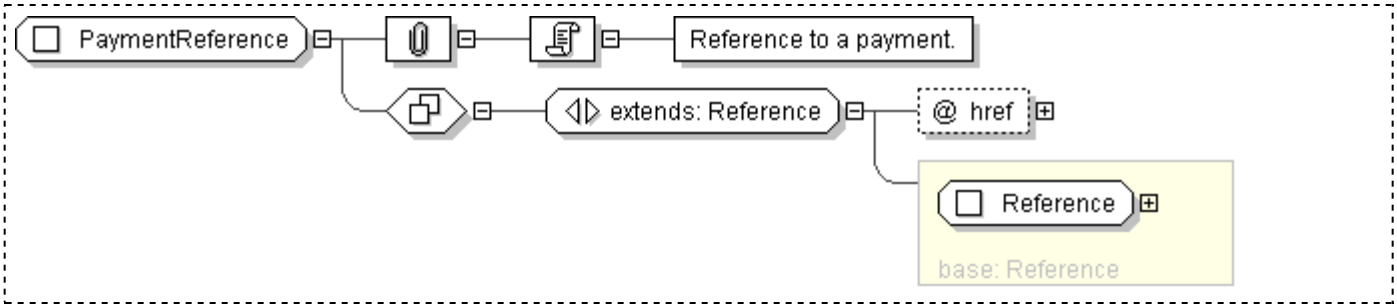
Super-types:	Reference < PaymentReference (by extension)
Sub-types:	None

Name	PaymentReference
Abstract	no
Documentation	Reference to a payment.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PaymentReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="PaymentBase"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PaymentType

[Table of contents]

Super-types:	xsd:normalizedString < Scheme (by restriction) < PaymentType (by extension)
Sub-types:	None

Name	PaymentType
Used by (from the same schema document)	Complex Type Payment
Abstract	no

XML Instance Representation

```
<...
 paymentTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="PaymentType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="paymentTypeScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Period

[Table of contents]

Super-types: None

Sub-types:

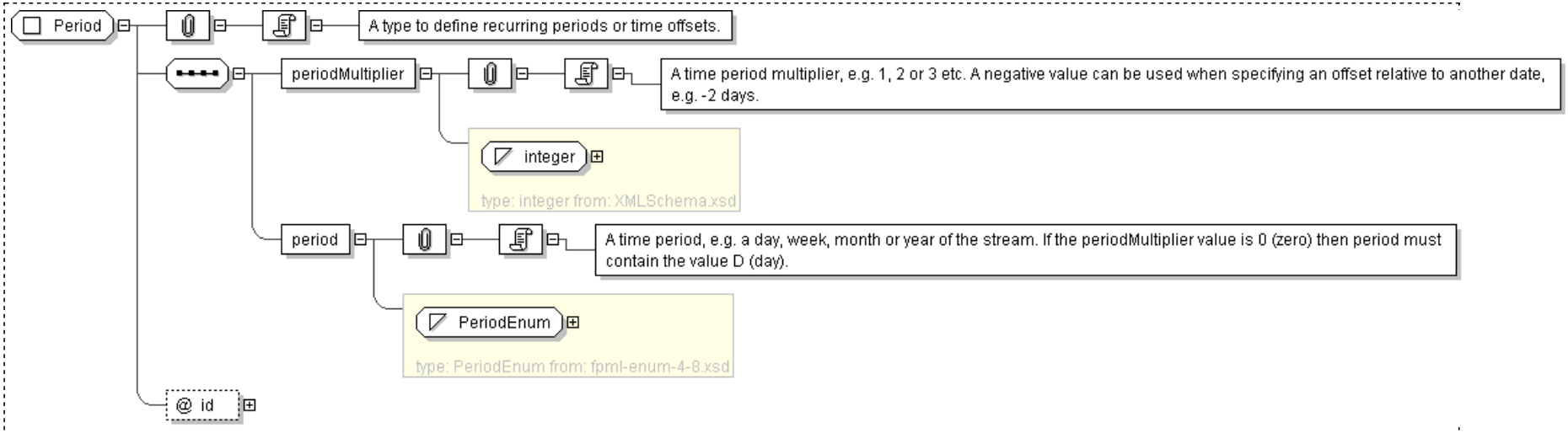
- Offset (by extension)
  - DateOffset (by extension)
  - RelativeDateOffset (by extension)
    - AdjustedRelativeDateOffset (by extension)
    - RelativeDates (by extension)

Name	Period
Used by (from the same schema document)	Complex Type ForecastRateIndex , Model Group FloatingRateIndex.model
Abstract	no
Documentation	A type to define recurring periods or time offsets.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to another date,
   e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0 (zero) then period must contain
   the value D (day).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Period">
   <xsd:sequence>
      <xsd:element name="periodMultiplier" type=" xsd:integer "/>
      <xsd:element name="period" type=" PeriodEnum "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PeriodicDates

[Table of contents]

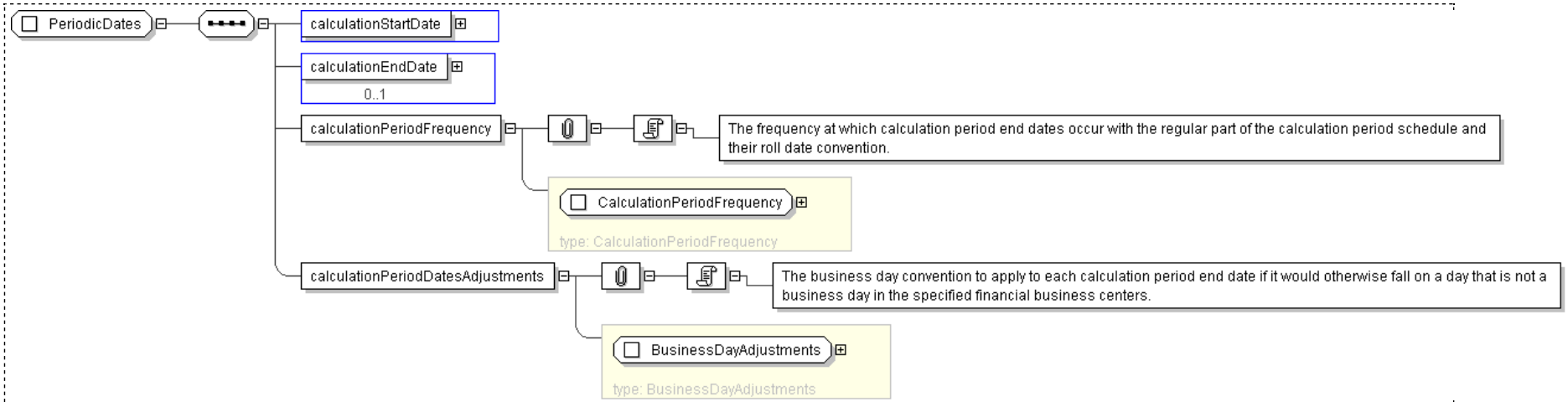
Super-types:	None
Sub-types:	None

Name	PeriodicDates
Used by (from the same schema document)	Complex Type AdjustableRelativeOrPeriodicDates , Complex Type AdjustableRelativeOrPeriodicDates2
Abstract	no

XML Instance Representation

```
<...>
   <calculationStartDate> AdjustableOrRelativeDate </calculationStartDate> [1]
   <calculationEndDate> AdjustableOrRelativeDate </calculationEndDate> [0..1]
   <calculationPeriodFrequency> CalculationPeriodFrequency </calculationPeriodFrequency> [1]
   'The frequency at which calculation period end dates occur with the regular part of the calculation period schedule and their roll date
   convention.'

   <calculationPeriodDatesAdjustments> BusinessDayAdjustments </calculationPeriodDatesAdjustments> [1]
   'The business day convention to apply to each calculation period end date if it would otherwise fall on a day that is not a business day in the
   specified financial business centers.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PeriodicDates">
   <xsd:sequence>
      <xsd:element name="calculationStartDate" type=" AdjustableOrRelativeDate "/>
      <xsd:element name="calculationEndDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      <xsd:element name="calculationPeriodFrequency" type=" CalculationPeriodFrequency "/>
      <xsd:element name="calculationPeriodDatesAdjustments" type=" BusinessDayAdjustments "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositiveAmountSchedule

[Table of contents]

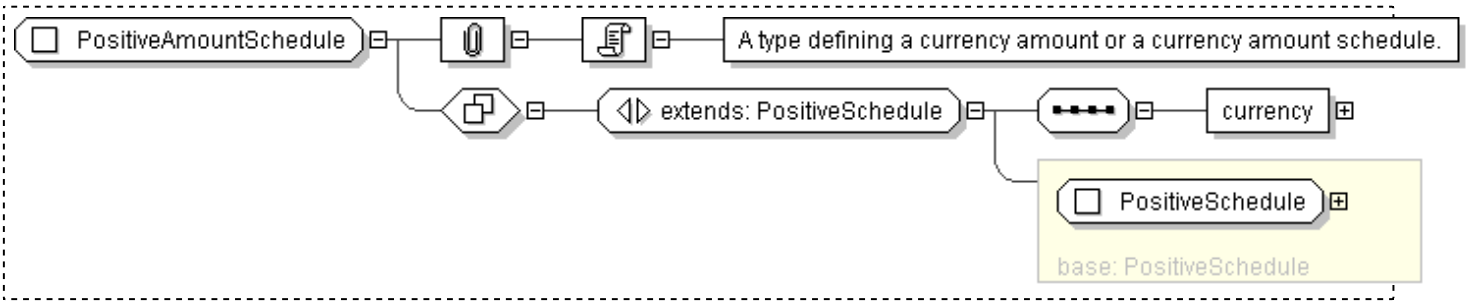
Super-types:	PositiveSchedule < PositiveAmountSchedule (by extension)
Sub-types:	None

Name	PositiveAmountSchedule
Abstract	no
Documentation	A type defining a currency amount or a currency amount schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> PositiveDecimal </initialValue> [1]
   'The strictly-positive initial rate or amount, as the case may be. An initial
   rate of 5% would be represented as 0.05.'

   <step> PositiveStep </step> [0..*]
   'The schedule of step date and strictly-positive value pairs. On each step date
   the associated step value becomes effective. A list of steps may be ordered in
   the document by ascending step date. An FpML document containing an unordered
   list of steps is still regarded as a conformant document.'

   <currency> Currency </currency> [1]
   'The currency in which an amount is denominated.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositiveAmountSchedule">
  <xsd:complexContent>
    <xsd:extension base=" PositiveSchedule ">
      <xsd:sequence>
        <xsd:element name="currency" type=" Currency "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

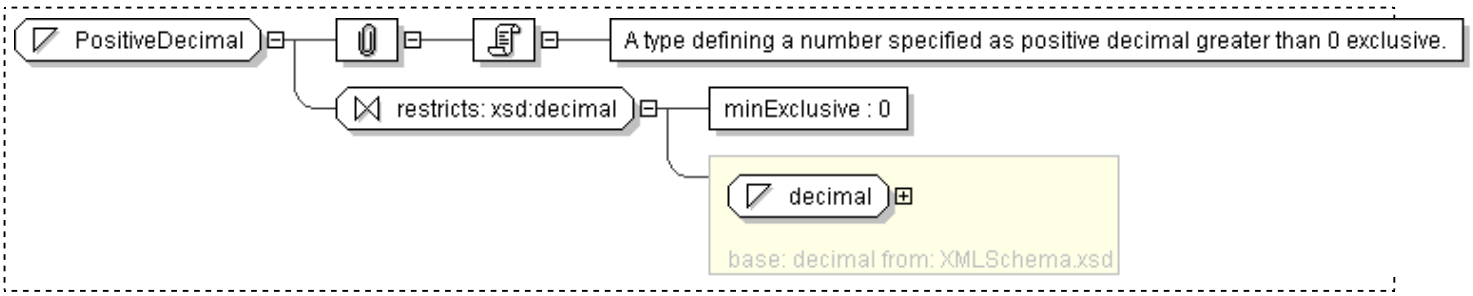
Simple Type: PositiveDecimal

[Table of contents]

Super-types:	xsd:decimal < PositiveDecimal (by restriction)
Sub-types:	None

Name	PositiveDecimal
Used by (from the same schema document)	Complex Type PositiveMoney , Complex Type PositiveSchedule , Complex Type PositiveStep
Content	• Base XSD Type: decimal • *value* > 0
Documentation	A type defining a number specified as positive decimal greater than 0 exclusive.

Diagram



Schema Component Representation

```
<xsd:simpleType name="PositiveDecimal">
  <xsd:restriction base=" xsd:decimal ">
    <xsd:minExclusive value="0"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositiveMoney

[Table of contents]

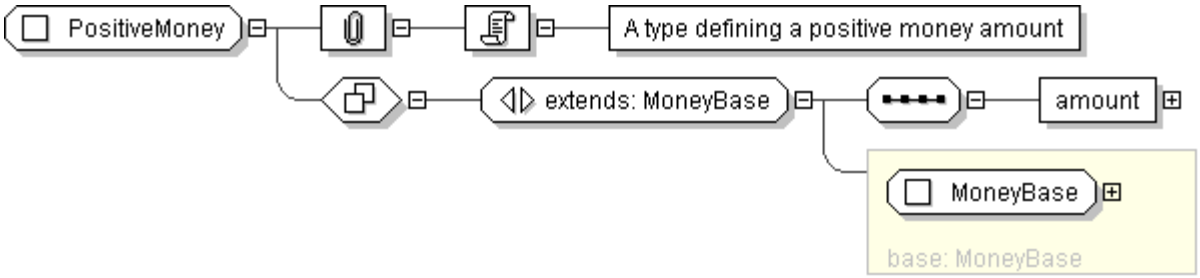
Super-types:	MoneyBase < PositiveMoney (by extension)
Sub-types:	None

Name	PositiveMoney
Used by (from the same schema document)	Complex Type PositivePayment
Abstract	no
Documentation	A type defining a positive money amount

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <currency> Currency </currency> [1]
    'The currency in which an amount is denominated.'

   <amount> PositiveDecimal </amount> [1]
    'The positive monetary quantity in currency units.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositiveMoney">
   <xsd:complexContent>
      <xsd:extension base=" MoneyBase ">
         <xsd:sequence>
            <xsd:element name="amount" type=" PositiveDecimal "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositivePayment

[Table of contents]

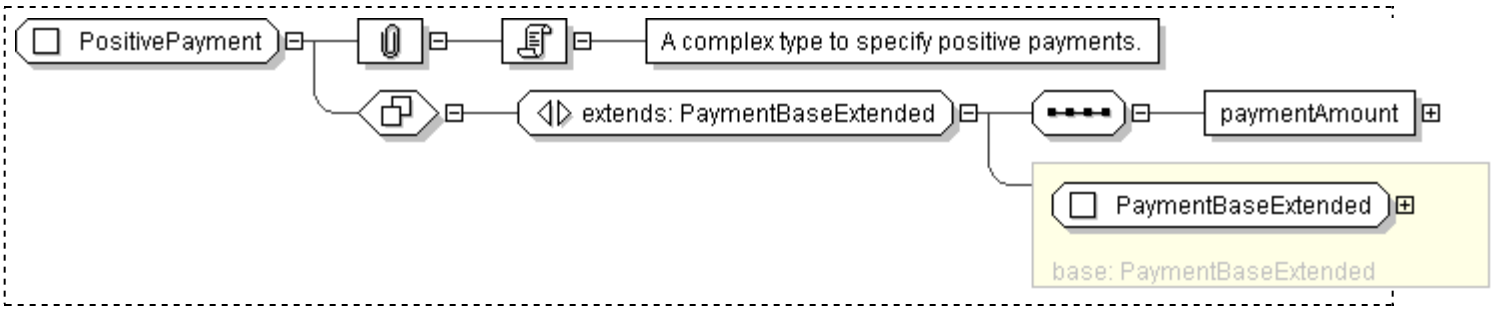
Super-types:	PaymentBase < PaymentBaseExtended (by extension) < PositivePayment (by extension)
Sub-types:	None

Name	PositivePayment
Abstract	no
Documentation	A complex type to specify positive payments.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
    'A reference to the party responsible for making the payments defined by this
    structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
    'A reference to the party that receives the payments corresponding to this
    structure.'

   <paymentDate> AdjustableOrRelativeDate </paymentDate> [1]
    'The payment date, which can be expressed as either an adjustable or relative
    date.'

   <paymentAmount> PositiveMoney </paymentAmount> [1]
    'Positive payment amount.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositivePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBaseExtended ">
         <xsd:sequence>
            <xsd:element name="paymentAmount" type=" PositiveMoney "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositiveSchedule

[Table of contents]

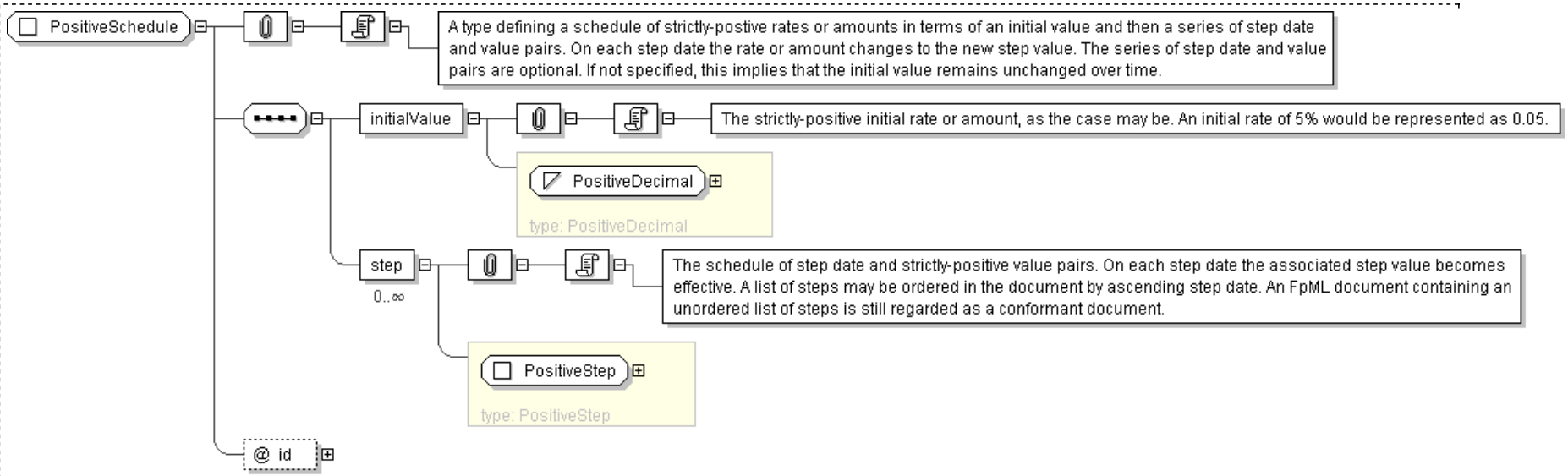
Super-types:	None
Sub-types:	• PositiveAmountSchedule (by extension)

Name	PositiveSchedule
Abstract	no
Documentation	A type defining a schedule of strictly-postive rates or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> PositiveDecimal </initialValue> [1]
   'The strictly-positive initial rate or amount, as the case may be. An initial rate of 5% would be represented as 0.05.'

   <step> PositiveStep </step> [0..*]
   'The schedule of step date and strictly-positive value pairs. On each step date the associated step value becomes effective. A
   list of steps may be ordered in the document by ascending step date. An FpML document containing an unordered list of steps is
   still regarded as a conformant document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositiveSchedule">
```

```
   <xsd:sequence>
      <xsd:element name="initialValue" type=" PositiveDecimal "/>
      <xsd:element name="step" type=" PositiveStep " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositiveStep

[Table of contents]

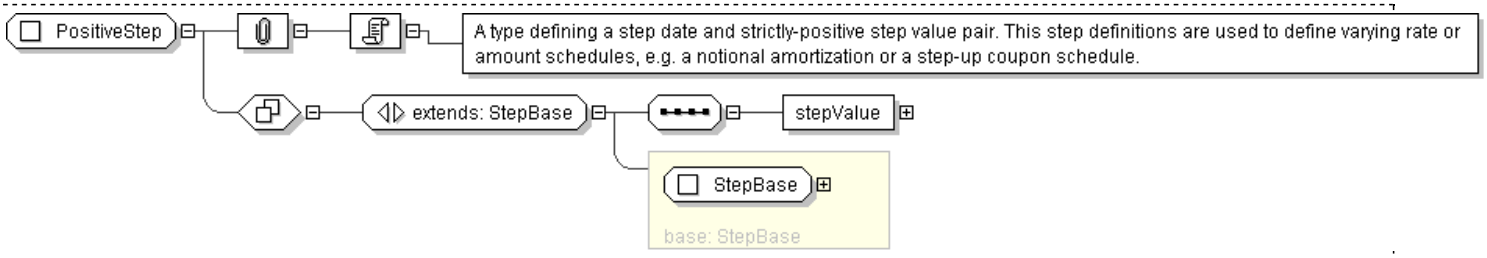
Super-types:	StepBase < PositiveStep (by extension)
Sub-types:	None

Name	PositiveStep
Used by (from the same schema document)	Complex Type PositiveSchedule
Abstract	no
Documentation	A type defining a step date and strictly-positive step value pair. This step definitions are used to define varying rate or amount schedules, e.g. a notional amortization or a step-up coupon schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <stepDate> xsd:date </stepDate> [1]
   'The date on which the associated stepValue becomes effective. This day may be subject to adjustment in
   accordance with a business day convention.'

   <stepValue> PositiveDecimal </stepValue> [1]
   'The strictly positive rate or amount which becomes effective on the associated stepDate. A rate of 5%
   would be represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositiveStep">
  <xsd:complexContent>
    <xsd:extension base=" StepBase ">
      <xsd:sequence>
        <xsd:element name="stepValue" type=" PositiveDecimal "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrevailingTime

[Table of contents]

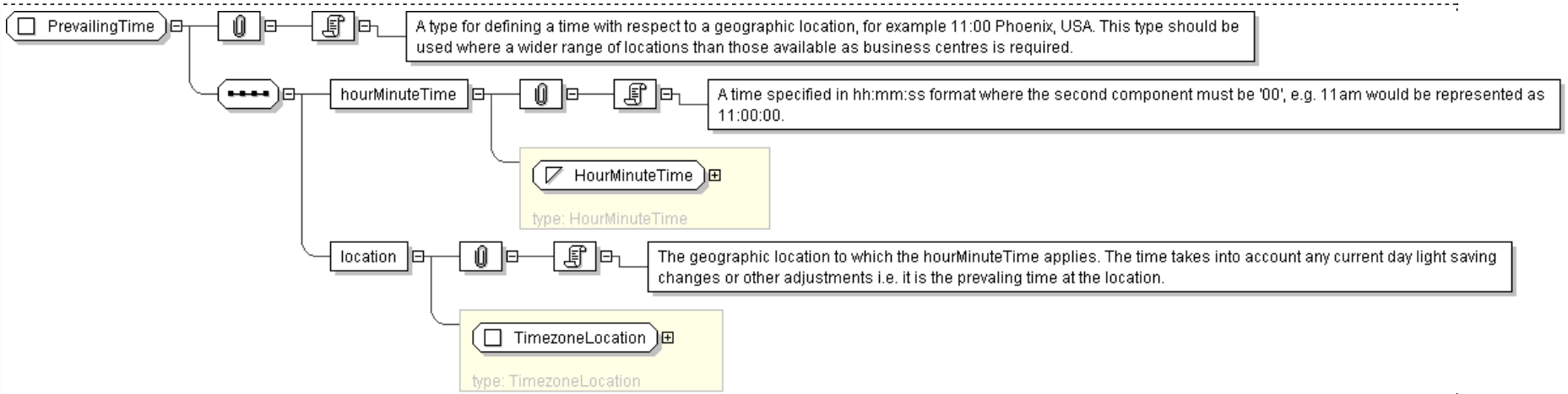
Super-types:	None
Sub-types:	None

Name	PrevailingTime
Used by (from the same schema document)	Complex Type OffsetPrevailingTime
Abstract	no
Documentation	A type for defining a time with respect to a geographic location, for example 11:00 Phoenix, USA. This type should be used where a wider range of locations than those available as business centres is required.

XML Instance Representation

```
<...>
   <hourMinuteTime> HourMinuteTime </hourMinuteTime> [1]
   'A time specified in hh:mm:ss format where the second component must be \'00\', e.g. 11am would be represented as 11:00:00.'

   <location> TimezoneLocation </location> [1]
   'The geographic location to which the hourMinuteTime applies. The time takes into account any current day light saving changes or
   other adjustments i.e. it is the prevaling time at the location.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrevailingTime">
   <xsd:sequence>
      <xsd:element name="hourMinuteTime" type=" HourMinuteTime "/>
      <xsd:element name="location" type=" TimezoneLocation "/>
   </xsd:sequence>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: PricingStructure

[Table of contents]

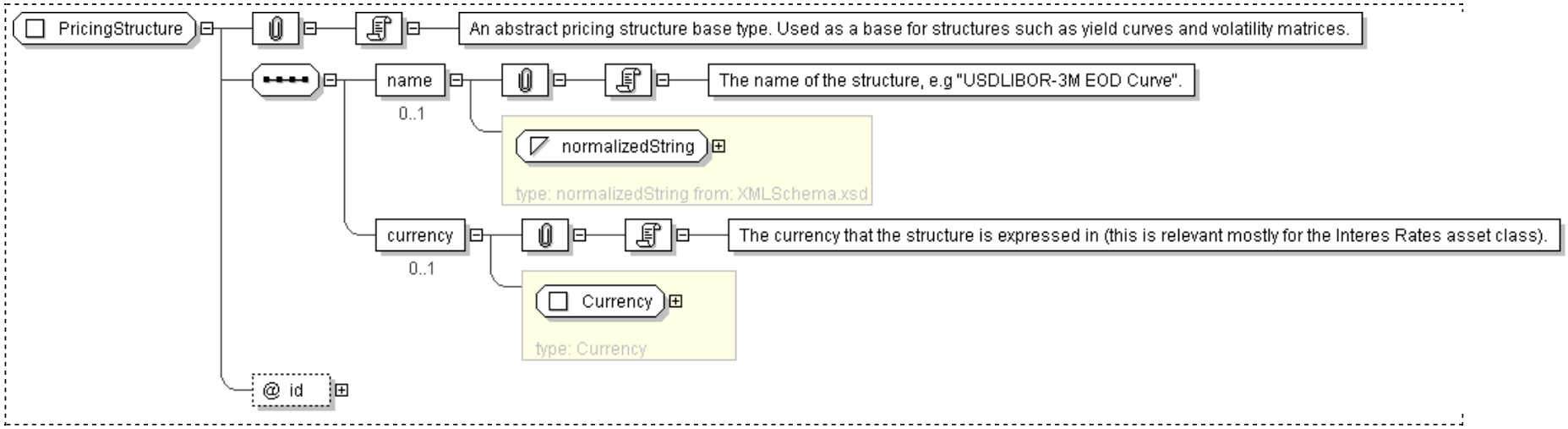
Super-types:	None
Sub-types:	None

Name	PricingStructure
Abstract	yes
Documentation	An abstract pricing structure base type. Used as a base for structures such as yield curves and volatility matrices.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:normalizedString </name> [0..1]
    'The name of the structure, e.g \"USDLIBOR-3M EOD Curve\".'

   <currency> Currency </currency> [0..1]
    'The currency that the structure is expressed in (this is relevant mostly for the Interes Rates asset class).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingStructure" abstract="true">
   <xsd:sequence>
      <xsd:element name="name" type=" xsd:normalizedString " minOccurs="0"/>
      <xsd:element name="currency" type=" Currency " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PricingStructureReference

[Table of contents]

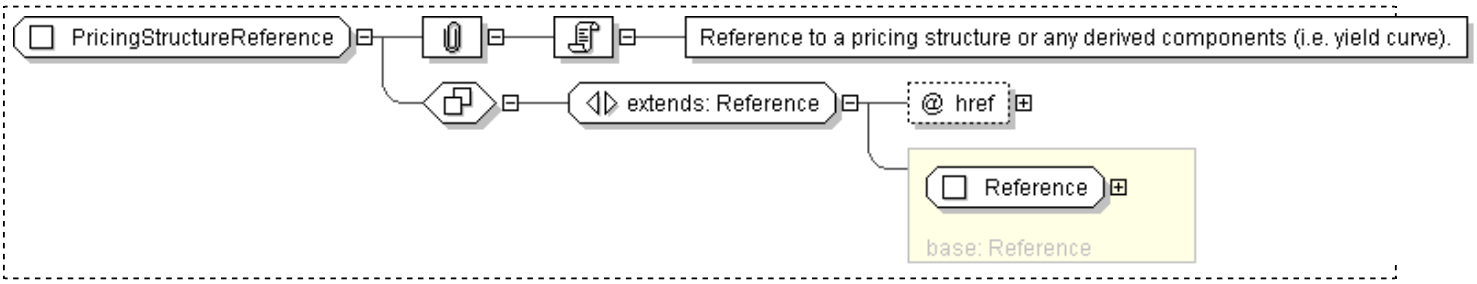
Super-types:	Reference < PricingStructureReference (by extension)
Sub-types:	None

Name	PricingStructureReference
Abstract	no
Documentation	Reference to a pricing structure or any derived components (i.e. yield curve).

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingStructureReference">
  <xsd:complexContent>
    <xsd:extension base=" Reference ">
      <xsd:attribute name="href" type=" xsd:IDREF " use="required"
      reference="PricingStructure"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PrincipalExchanges

[Table of contents]

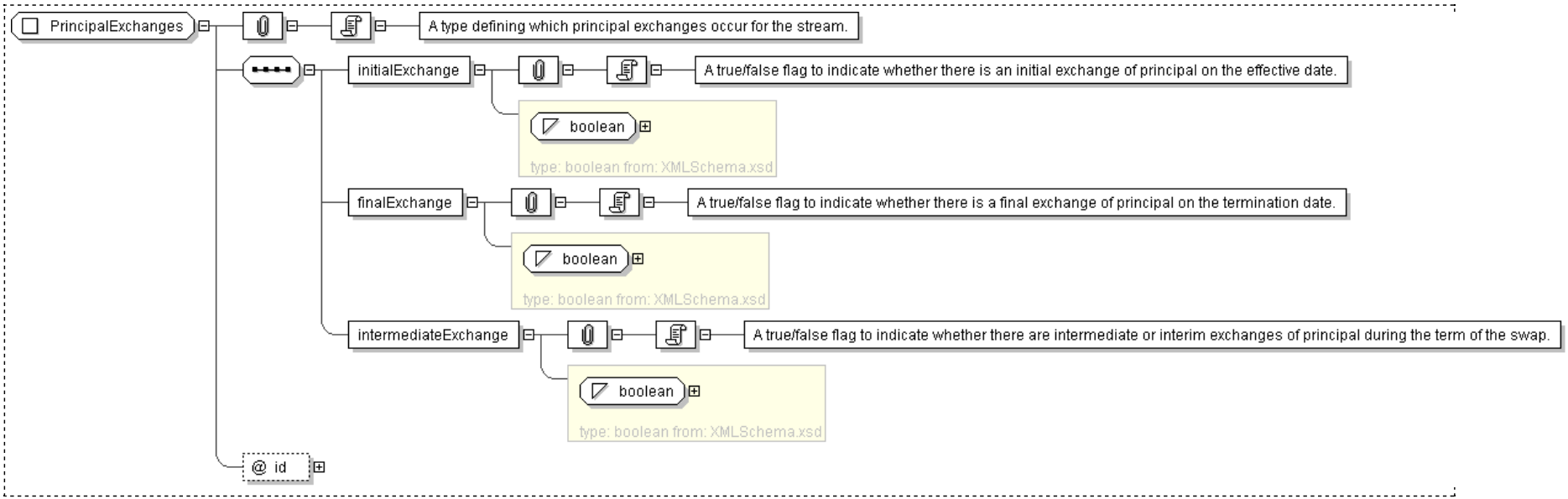
Super-types:	None
Sub-types:	None

Name	PrincipalExchanges
Abstract	no
Documentation	A type defining which principal exchanges occur for the stream.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialExchange> xsd:boolean </initialExchange> [1]
    'A true/false flag to indicate whether there is an initial exchange of principal on the effective date.'

   <finalExchange> xsd:boolean </finalExchange> [1]
    'A true/false flag to indicate whether there is a final exchange of principal on the termination date.'

   <intermediateExchange> xsd:boolean </intermediateExchange> [1]
    'A true/false flag to indicate whether there are intermediate or interim exchanges of principal during the term of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchanges">
   <xsd:sequence>
      <xsd:element name="initialExchange" type=" xsd:boolean "/>
      <xsd:element name="finalExchange" type=" xsd:boolean "/>
      <xsd:element name="intermediateExchange" type=" xsd:boolean "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="optional"/>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Product

[Table of contents]

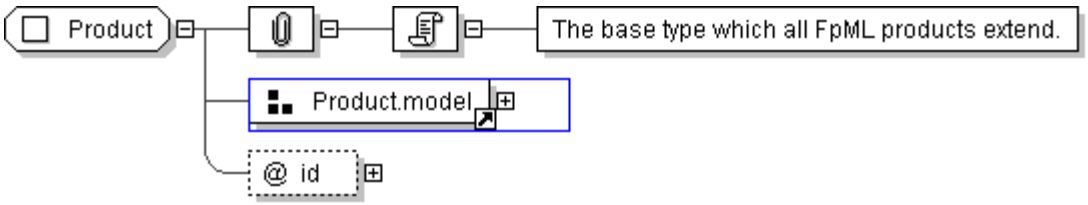
Super-types:	None
Sub-types:	None

Name	Product
Used by (from the same schema document)	Element product
Abstract	yes
Documentation	The base type which all FpML products extend.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product
    categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the
    domain values associated with this element. Note that the domain values for
    this element are not strictly an enumerated list.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Product" abstract="true">
   <xsd:group ref=" Product.model "/>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProductId

[Table of contents]

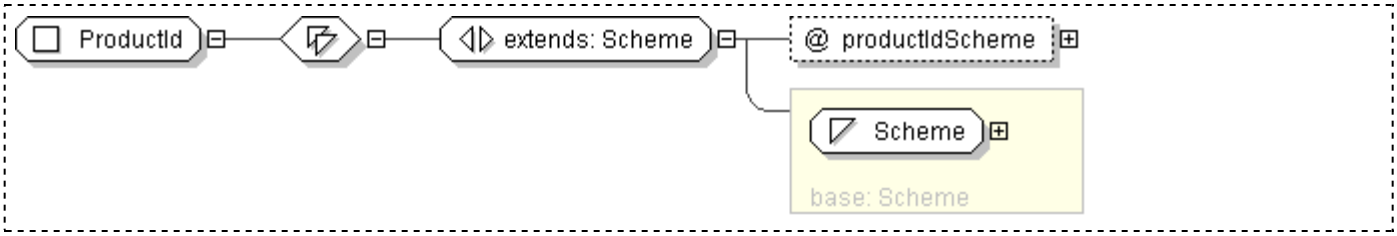
Super-types:	xsd:normalizedString < Scheme (by restriction) < ProductId (by extension)
Sub-types:	None

Name	ProductId
Used by (from the same schema document)	Model Group Product.model
Abstract	no

XML Instance Representation

```
<...
 productIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProductId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="productIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProductReference

[Table of contents]

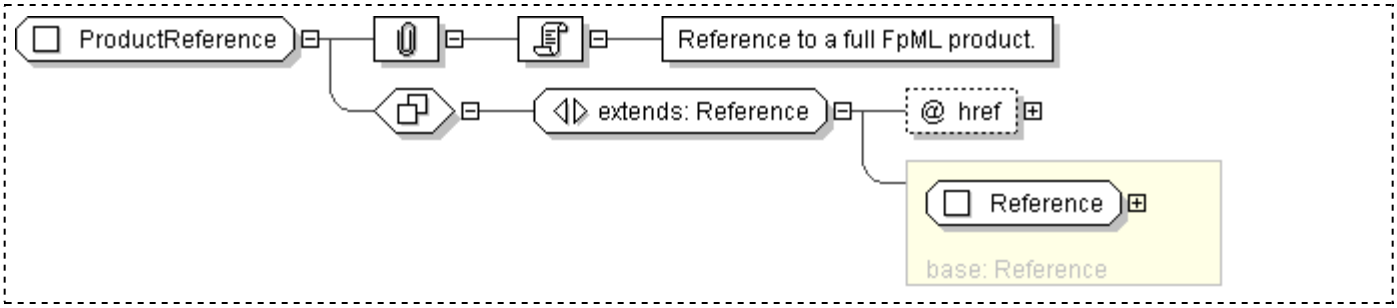
Super-types:	Reference < ProductReference (by extension)
Sub-types:	None

Name	ProductReference
Abstract	no
Documentation	Reference to a full FpML product.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProductReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Product"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ProductType

[Table of contents]

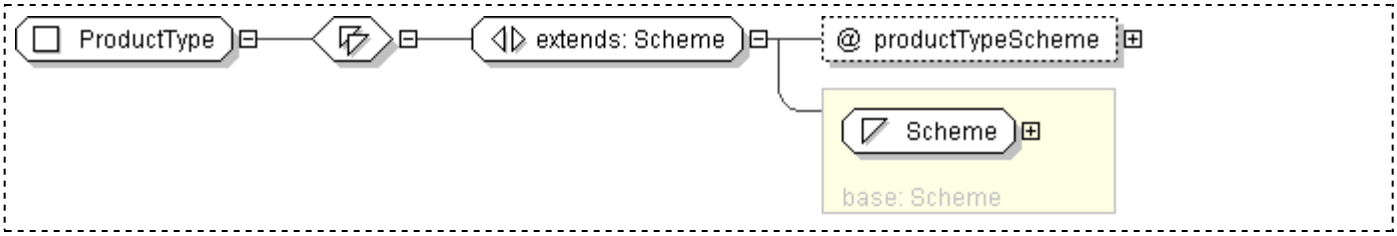
Super-types:	xsd:normalizedString < Scheme (by restriction) < ProductType (by extension)
Sub-types:	None

Name	ProductType
Used by (from the same schema document)	Model Group Product.model
Abstract	no

XML Instance Representation

```
<...
 productTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ProductType">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="productTypeScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/product-type-simple"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: QuotedCurrencyPair

[Table of contents]

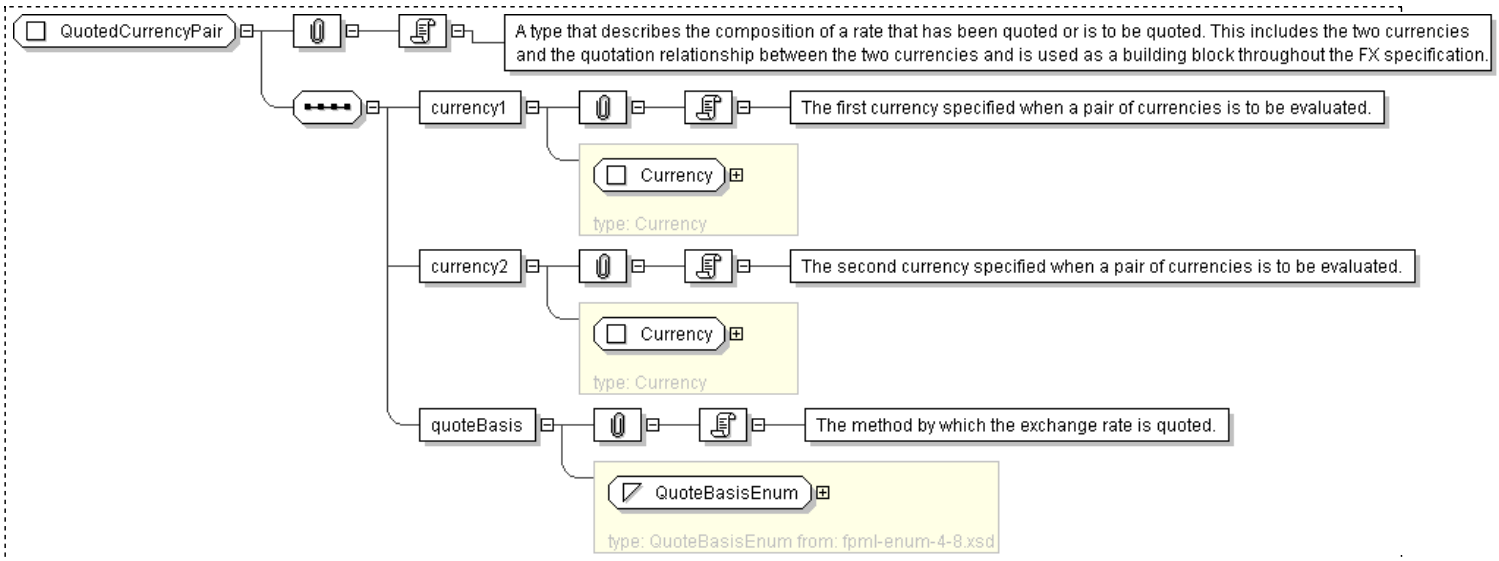
Super-types:	None
Sub-types:	None

Name	QuotedCurrencyPair
Used by (from the same schema document)	Complex Type FxFixing , Complex Type FxRate
Abstract	no
Documentation	A type that describes the composition of a rate that has been quoted or is to be quoted. This includes the two currencies and the quotation relationship between the two currencies and is used as a building block throughout the FX specification.

XML Instance Representation

```
<...>
  <currency1> Currency </currency1> [1]
  'The first currency specified when a pair of currencies is to be evaluated.'

  <currency2> Currency </currency2> [1]
  'The second currency specified when a pair of currencies is to be evaluated.'

  <quoteBasis> QuoteBasisEnum </quoteBasis> [1]
  'The method by which the exchange rate is quoted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="QuotedCurrencyPair">
  <xsd:sequence>
    <xsd:element name="currency1" type=" Currency "/>
    <xsd:element name="currency2" type=" Currency "/>
    <xsd:element name="quoteBasis" type=" QuoteBasisEnum "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Rate

[Table of contents]

Super-types: None

Sub-types:

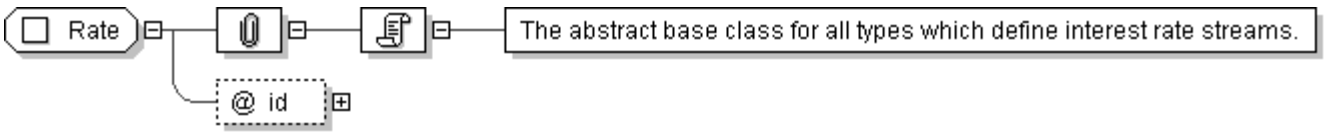
- FloatingRate (by extension)
  - FloatingRateCalculation (by extension)

Name	Rate
Abstract	yes
Documentation	The abstract base class for all types which define interest rate streams.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Rate" abstract="true">
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RateObservation

[Table of contents]

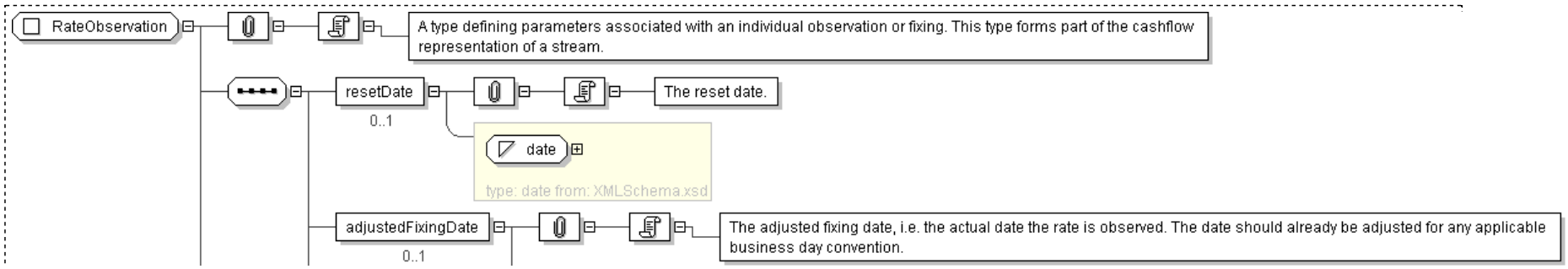
Super-types:	None
Sub-types:	None

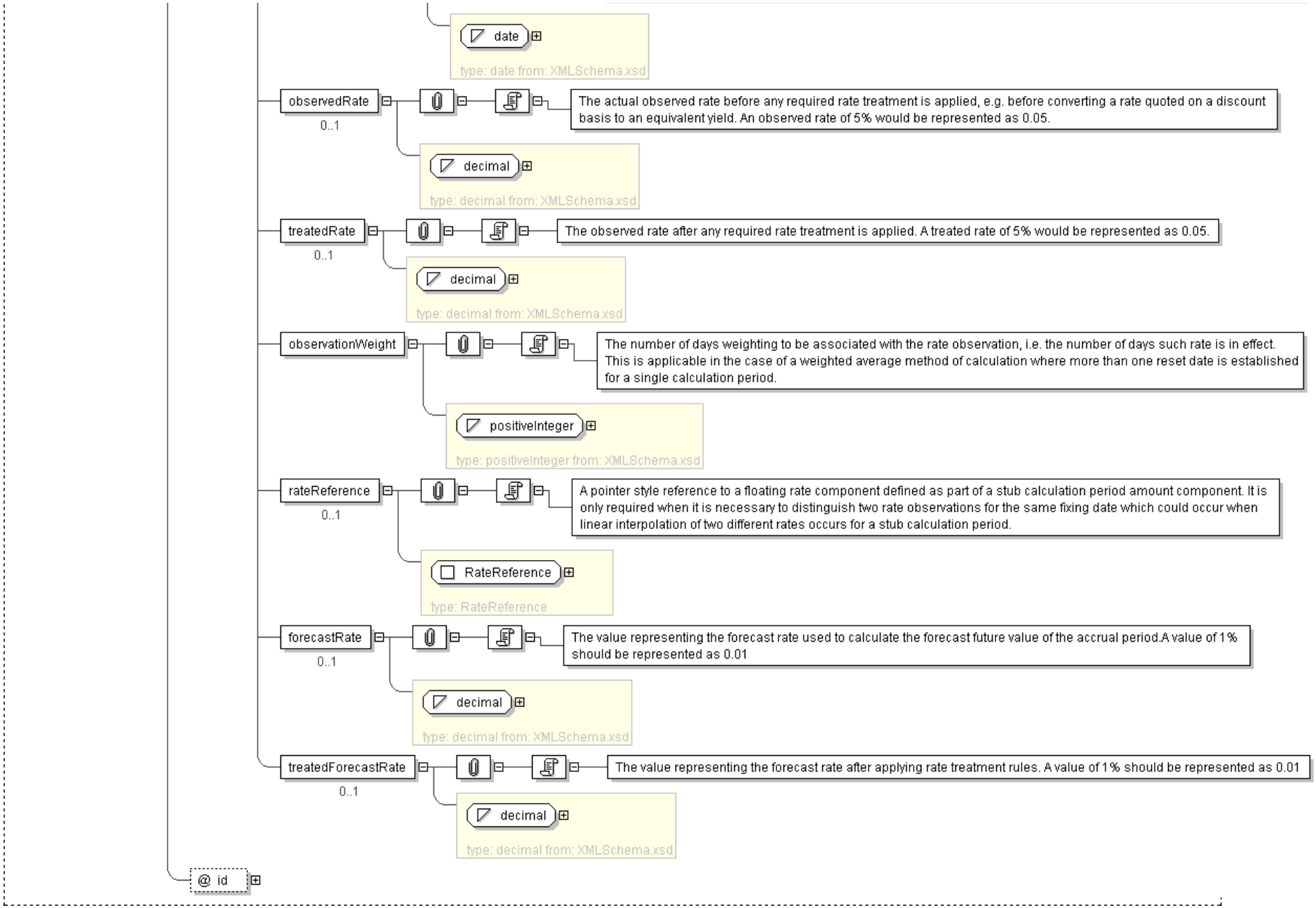
Name	RateObservation
Abstract	no
Documentation	A type defining parameters associated with an individual observation or fixing. This type forms part of the cashflow representation of a stream.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <resetDate> xsd:date </resetDate> [0..1]
   'The reset date.'

   <adjustedFixingDate> xsd:date </adjustedFixingDate> [0..1]
   'The adjusted fixing date, i.e. the actual date the rate is observed. The date should already be adjusted for any applicable business day
   convention.'

   <observedRate> xsd:decimal </observedRate> [0..1]
   'The actual observed rate before any required rate treatment is applied, e.g. before converting a rate quoted on a discount basis to an
   equivalent yield. An observed rate of 5% would be represented as 0.05.'

   <treatedRate> xsd:decimal </treatedRate> [0..1]
   'The observed rate after any required rate treatment is applied. A treated rate of 5% would be represented as 0.05.'

   <observationWeight> xsd:positiveInteger </observationWeight> [1]
   'The number of days weighting to be associated with the rate observation, i.e. the number of days such rate is in effect. This is
   applicable in the case of a weighted average method of calculation where more than one reset date is established for a single calculation
   period.'

   <rateReference> RateReference </rateReference> [0..1]
   'A pointer style reference to a floating rate component defined as part of a stub calculation period amount component. It is only required
   when it is necessary to distinguish two rate observations for the same fixing date which could occur when linear interpolation of two
   different rates occurs for a stub calculation period.'

   <forecastRate> xsd:decimal </forecastRate> [0..1]
   'The value representing the forecast rate used to calculate the forecast future value of the accrual period.A value of 1% should be
   represented as 0.01'

   <treatedForecastRate> xsd:decimal </treatedForecastRate> [0..1]
   'The value representing the forecast rate after applying rate treatment rules. A value of 1% should be represented as 0.01'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="RateObservation">
  <xsd:sequence>
    <xsd:element name="resetDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="adjustedFixingDate" type=" xsd:date " minOccurs="0"/>
    <xsd:element name="observedRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="treatedRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="observationWeight" type=" xsd:positiveInteger "/>
    <xsd:element name="rateReference" type=" RateReference " minOccurs="0"/>
    <xsd:element name="forecastRate" type=" xsd:decimal " minOccurs="0"/>
    <xsd:element name="treatedForecastRate" type=" xsd:decimal " minOccurs="0"/>
```

```
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RateReference

[Table of contents]

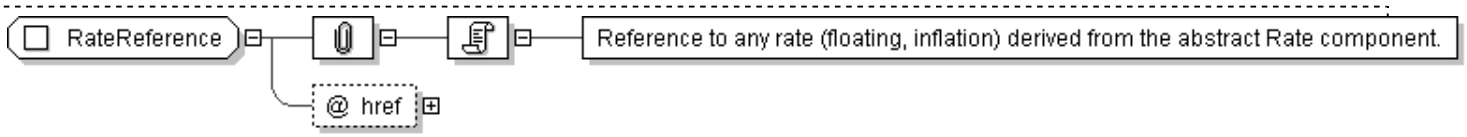
Super-types:	None
Sub-types:	None

Name	RateReference
Used by (from the same schema document)	Complex Type RateObservation
Abstract	no
Documentation	Reference to any rate (floating, inflation) derived from the abstract Rate component.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RateReference">
   <xsd:attribute name="href" type=" xsd:IDREF " use="required" reference="Rate"/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RateSourcePage

[Table of contents]

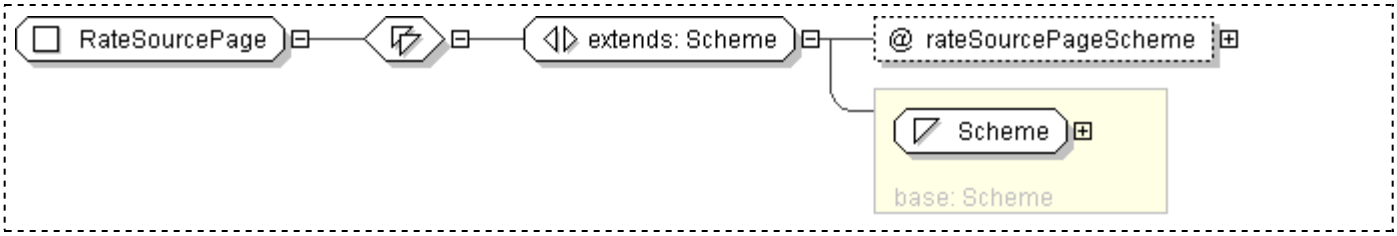
Super-types:	xsd:normalizedString < Scheme (by restriction) < RateSourcePage (by extension)
Sub-types:	None

Name	RateSourcePage
Used by (from the same schema document)	Complex Type InformationSource
Abstract	no

XML Instance Representation

```
<...
 rateSourcePageScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RateSourcePage">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="rateSourcePageScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Reference

[Table of contents]

Super-types: None

Sub-types:

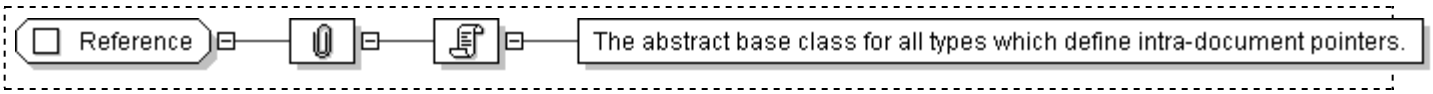
- AccountReference (by extension)
- AmountReference (by extension)
- BusinessCentersReference (by extension)
- BusinessDayAdjustmentsReference (by extension)
- DateReference (by extension)
- DeterminationMethodReference (by extension)
- IdentifiedCurrencyReference (by extension)
- LegalEntityReference (by extension)
- NotionalAmountReference (by extension)
- NotionalReference (by extension)
- PartyOrAccountReference (by extension)
- PartyOrTradeSideReference (by extension)
- PartyReference (by extension)
- PaymentReference (by extension)
- PricingStructureReference (by extension)
- ProductReference (by extension)
- ReturnSwapNotionalAmountReference (by extension)
- ScheduleReference (by extension)
- SpreadScheduleReference (by extension)

Name	Reference
Abstract	yes
Documentation	The abstract base class for all types which define intra-document pointers.

XML Instance Representation

```
<.../>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Reference" abstract="true"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceAmount

[Table of contents]

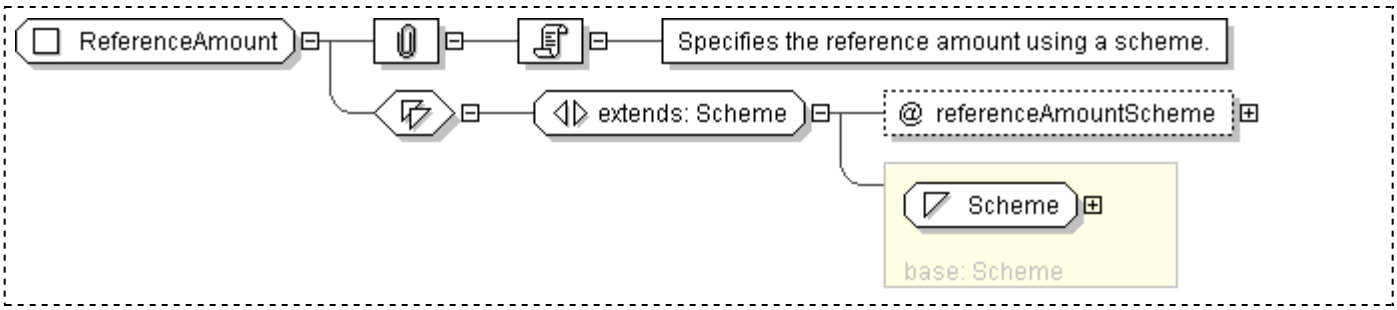
Super-types:	xsd:normalizedString < Scheme (by restriction) < ReferenceAmount (by extension)
Sub-types:	None

Name	ReferenceAmount
Abstract	no
Documentation	Specifies the reference amount using a scheme.

XML Instance Representation

```
<...
 referenceAmountScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferenceAmount">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="referenceAmountScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceBank

[Table of contents]

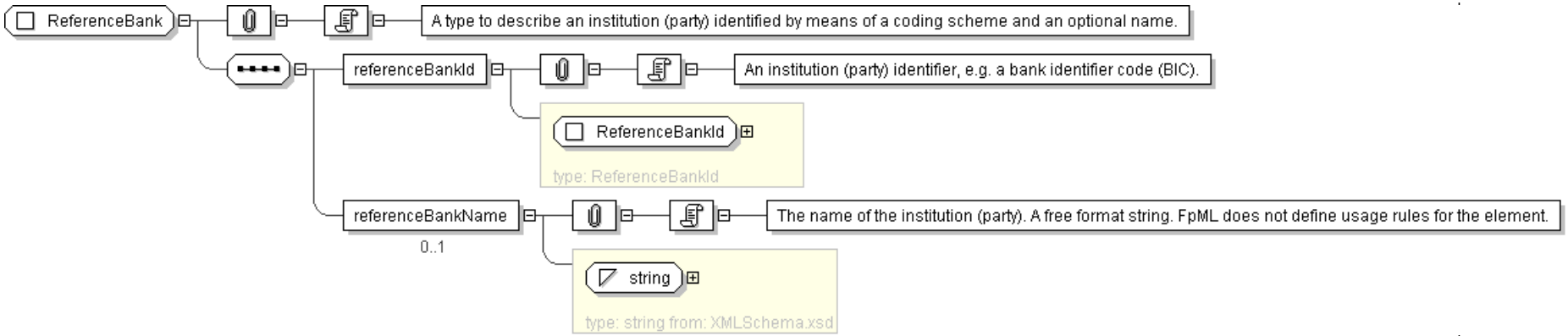
Super-types:	None
Sub-types:	None

Name	ReferenceBank
Used by (from the same schema document)	Complex Type CashSettlementReferenceBanks
Abstract	no
Documentation	A type to describe an institution (party) identified by means of a coding scheme and an optional name.

XML Instance Representation

```
<...>
  <referenceBankId> ReferenceBankId </referenceBankId> [1]
  'An institution (party) identifier, e.g. a bank identifier code (BIC).'

  <referenceBankName> xsd:string </referenceBankName> [0..1]
  'The name of the institution (party). A free format string. FpML does not define usage rules for the element.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferenceBank">
  <xsd:sequence>
    <xsd:element name="referenceBankId" type=" ReferenceBankId "/>
    <xsd:element name="referenceBankName" type=" xsd:string " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReferenceBankId

[Table of contents]

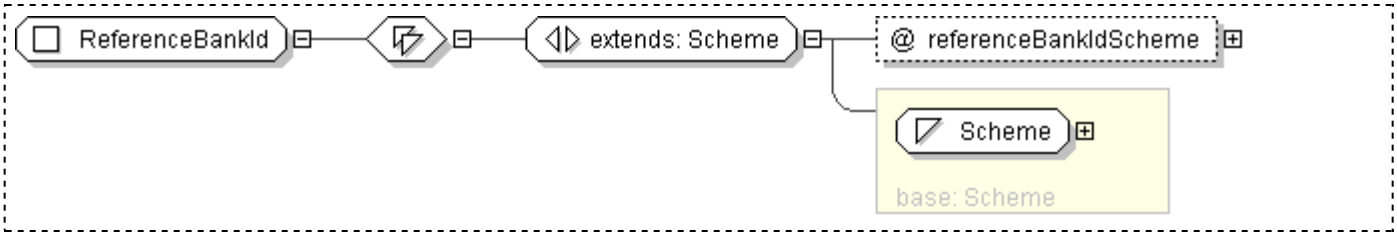
Super-types:	xsd:normalizedString < Scheme (by restriction) < ReferenceBankId (by extension)
Sub-types:	None

Name	ReferenceBankId
Used by (from the same schema document)	Complex Type ReferenceBank
Abstract	no

XML Instance Representation

```
<...
 referenceBankIdScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReferenceBankId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="referenceBankIdScheme" type=" xsd:anyURI "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RelativeDateOffset

[Table of contents]

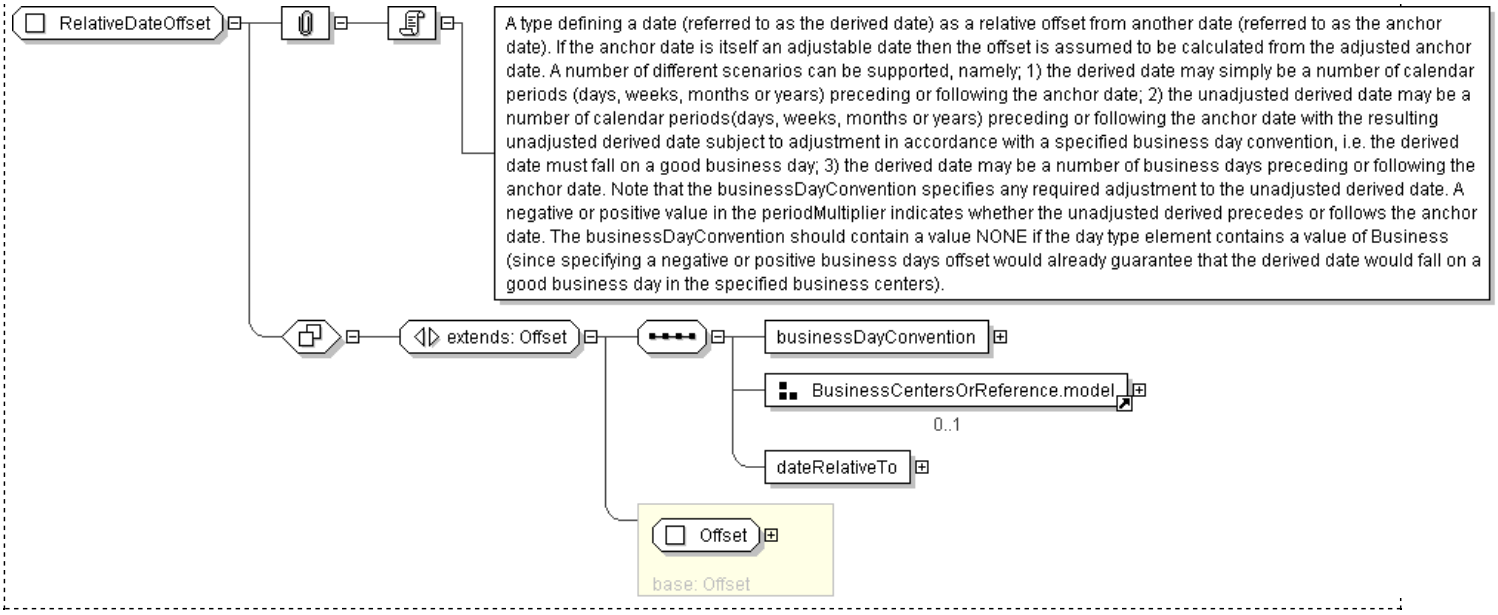
Super-types:	Period < Offset (by extension) < RelativeDateOffset (by extension)
Sub-types:	• AdjustedRelativeDateOffset (by extension) • RelativeDates (by extension)

Name	RelativeDateOffset
Used by (from the same schema document)	Complex Type AdjustableDatesOrRelativeDateOffset , Complex Type AdjustableOrRelativeDate , Complex Type ExerciseFee , Complex Type ExerciseFeeSchedule
Abstract	no
Documentation	A type defining a date (referred to as the derived date) as a relative offset from another date (referred to as the anchor date). If the anchor date is itself an adjustable date then the offset is assumed to be calculated from the adjusted anchor date. A number of different scenarios can be supported, namely; 1) the derived date may simply be a number of calendar periods (days, weeks, months or years) preceding or following the anchor date; 2) the unadjusted derived date may be a number of calendar periods(days, weeks, months or years) preceding or following the anchor date with the resulting unadjusted derived date subject to adjustment in accordance with a specified business day convention, i.e. the derived date must fall on a good business day; 3) the derived date may be a number of business days preceding or following the anchor date. Note that the businessDayConvention specifies any required adjustment to the unadjusted derived date. A negative or positive value in the periodMultiplier indicates whether the unadjusted derived precedes or follows the anchor date. The businessDayConvention should contain a value NONE if the day type element contains a value of Business (since specifying a negative or positive business days offset would already guarantee that the derived date would fall on a good business day in the specified business centers).

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative
   to another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the periodMultiplier value is 0 (zero) then
   period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
   'In the case of an offset specified as a number of days, this element defines whether consideration is given as
   to whether a day is a good business day or not. If a day type of business days is specified then non-business
   days are ignored when calculating the offset. The financial business centers to use for determination of
   business days are implied by the context in which this element is used. This element must only be included when
   the offset is specified as a number of days. If the offset is zero days then the dayType element should not be
   included.'

   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
   'The convention for adjusting a date if it would otherwise fall on a day that is not a business day.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
         'A pointer style reference to a set of financial business centers defined elsewhere in the document. This
         set of business centers is used to determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
   <dateRelativeTo> DateReference </dateRelativeTo> [1]
   'Specifies the anchor as an href attribute. The href attribute value is a pointer style reference to the
   element or component elsewhere in the document where the anchor date is defined.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RelativeDateOffset">
  <xsd:complexContent>
    <xsd:extension base=" Offset ">
      <xsd:sequence>
        <xsd:element name="businessDayConvention" type=" BusinessDayConventionEnum "/>
        <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
        <xsd:element name="dateRelativeTo" type=" DateReference "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RelativeDates

[Table of contents]

Super-types:	Period < Offset (by extension) < RelativeDateOffset (by extension) < RelativeDates (by extension)
Sub-types:	None

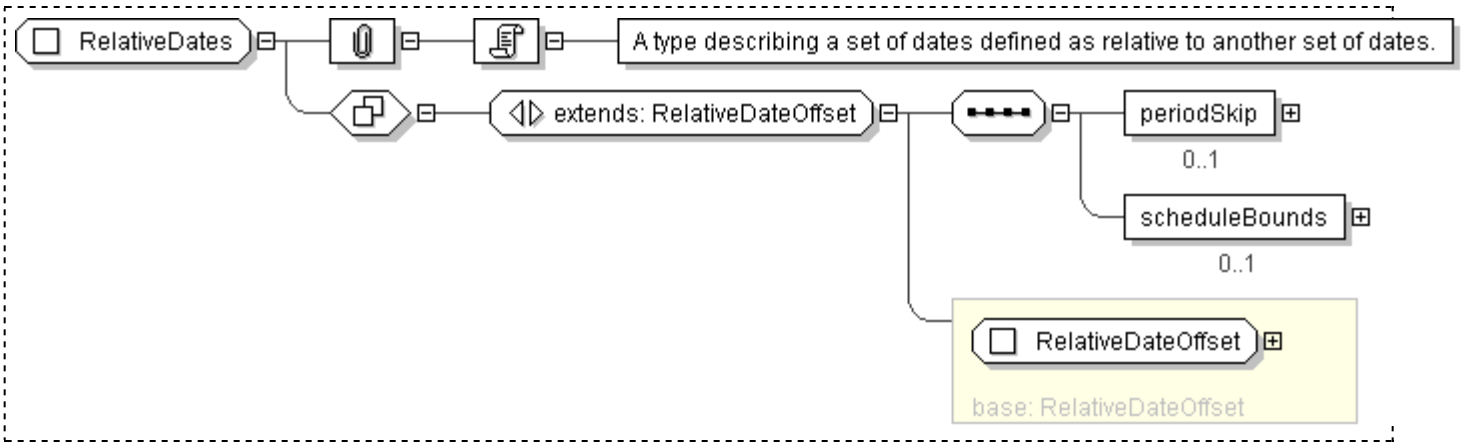
Name	RelativeDates
Used by (from the same schema document)	Complex Type AdjustableOrRelativeDates , Complex Type AdjustableRelativeOrPeriodicDates2
Abstract	no
Documentation	A type describing a set of dates defined as relative to another set of dates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used
   when specifying an offset relative to another date, e.g. -2 days.'

   <period> PeriodEnum </period> [1]
   'A time period, e.g. a day, week, month or year of the stream. If the
   periodMultiplier value is 0 (zero) then period must contain the value D (day).'

   <dayType> DayTypeEnum </dayType> [0..1]
   'In the case of an offset specified as a number of days, this element defines
   whether consideration is given as to whether a day is a good business day or
   not. If a day type of business days is specified then non-business days are
   ignored when calculating the offset. The financial business centers to use for
   determination of business days are implied by the context in which this element
   is used. This element must only be included when the offset is specified as a
   number of days. If the offset is zero days then the dayType element should not
   be included.'

   <businessDayConvention> BusinessDayConventionEnum </businessDayConvention> [1]
   'The convention for adjusting a date if it would otherwise fall on a day that
   is not a business day.'

   Start Group: BusinessCentersOrReference.model [0..1]
      Start Choice [1]
         <businessCentersReference> BusinessCentersReference </businessCentersReference>
         [1]
         'A pointer style reference to a set of financial business centers defined
         elsewhere in the document. This set of business centers is used to
         determine whether a particular day is a business day or not.'

         <businessCenters> BusinessCenters </businessCenters> [1]
      End Choice
   End Group: BusinessCentersOrReference.model
   <dateRelativeTo> DateReference </dateRelativeTo> [1]
   'Specifies the anchor as an href attribute. The href attribute value is a
   pointer style reference to the element or component elsewhere in the document
   where the anchor date is defined.'

   <periodSkip> xsd:positiveInteger </periodSkip> [0..1]
   'The number of periods in the referenced date schedule that are between each
   date in the relative date schedule. Thus a skip of 2 would mean that dates are
   relative to every second date in the referenced schedule. If present this
   should have a value greater than 1.'
```

```
<scheduleBounds> DateRange </scheduleBounds> [0..1]
'The first and last dates of a schedule. This can be used to restrict the range
of values in a reference series of dates.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RelativeDates">
   <xsd:complexContent>
      <xsd:extension base=" RelativeDateOffset ">
         <xsd:sequence>
            <xsd:element name="periodSkip" type=" xsd:positiveInteger " minOccurs="0"/>
            <xsd:element name="scheduleBounds" type=" DateRange " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RelativeDateSequence

[Table of contents]

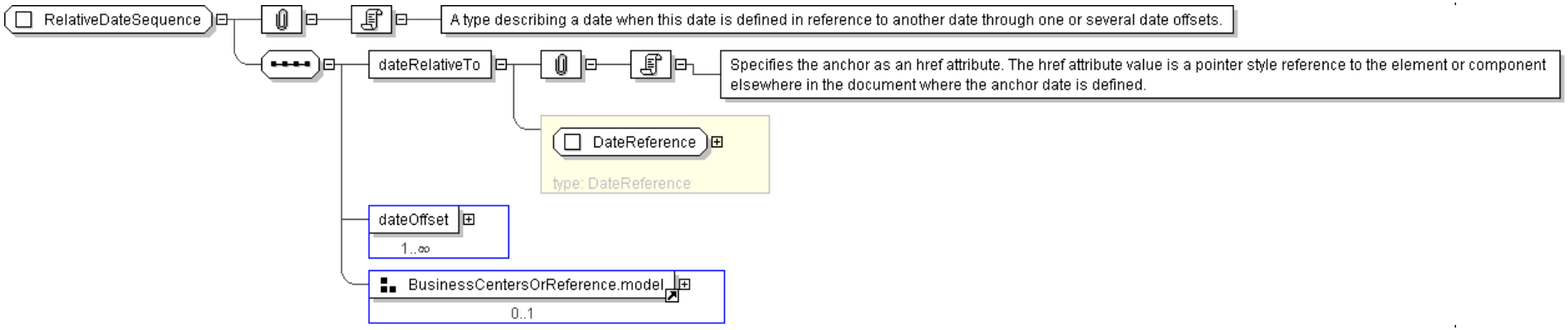
Super-types:	None
Sub-types:	None

Name	RelativeDateSequence
Used by (from the same schema document)	Complex Type AdjustableRelativeOrPeriodicDates
Abstract	no
Documentation	A type describing a date when this date is defined in reference to another date through one or several date offsets.

XML Instance Representation

```
<...>
  <dateRelativeTo> DateReference </dateRelativeTo> [1]
  'Specifies the anchor as an href attribute. The href attribute value is a pointer style reference to the element or component elsewhere in
  the document where the anchor date is defined.'

  <dateOffset> DateOffset </dateOffset> [1..*]
  Start Group: BusinessCentersOrReference.model [0..1]
    Start Choice [1]
      <businessCentersReference> BusinessCentersReference </businessCentersReference> [1]
      'A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is
      used to determine whether a particular day is a business day or not.'

      <businessCenters> BusinessCenters </businessCenters> [1]
    End Choice
  End Group: BusinessCentersOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RelativeDateSequence">
  <xsd:sequence>
    <xsd:element name="dateRelativeTo" type=" DateReference "/>
    <xsd:element name="dateOffset" type=" DateOffset " maxOccurs="unbounded"/>
    <xsd:group ref=" BusinessCentersOrReference.model " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequiredIdentifierDate

[Table of contents]

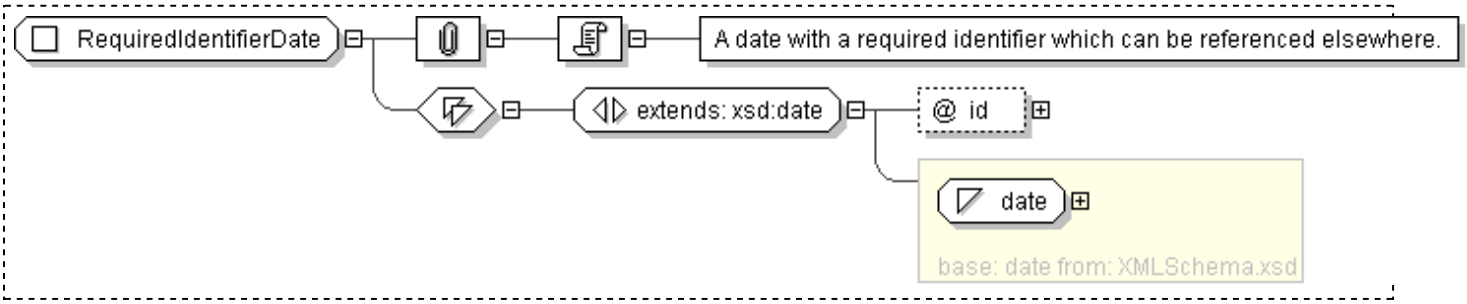
Super-types:	xsd:date < RequiredIdentifierDate (by extension)
Sub-types:	None

Name	RequiredIdentifierDate
Abstract	no
Documentation	A date with a required identifier which can be referenced elsewhere.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
   xsd:date
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequiredIdentifierDate">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:date ">
         <xsd:attribute name="id" type=" xsd:ID " use="required"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ResetFrequency

[Table of contents]

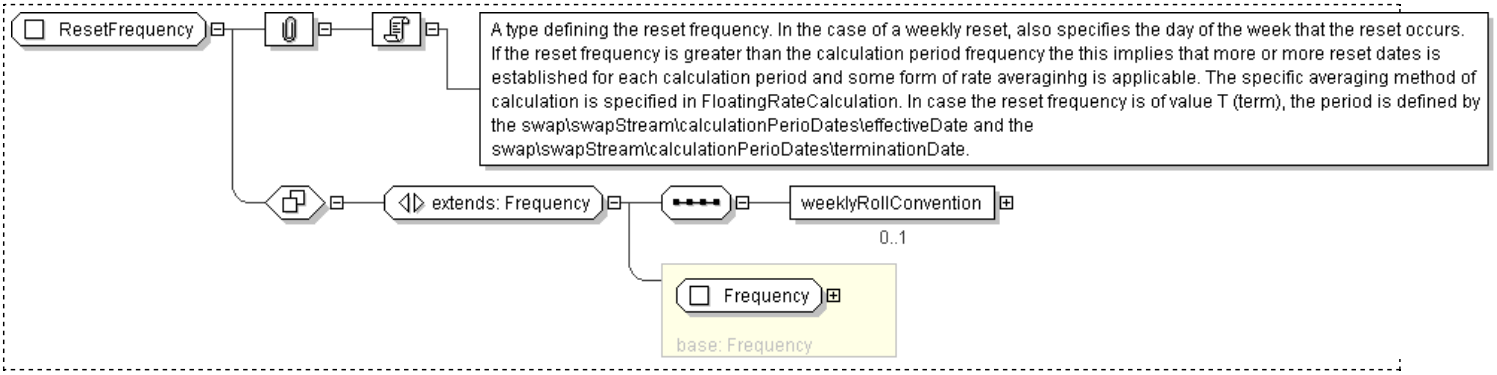
Super-types:	Frequency < ResetFrequency (by extension)
Sub-types:	None

Name	ResetFrequency
Abstract	no
Documentation	A type defining the reset frequency. In the case of a weekly reset, also specifies the day of the week that the reset occurs. If the reset frequency is greater than the calculation period frequency the this implies that more or more reset dates is established for each calculation period and some form of rate averaginhg is applicable. The specific averaging method of calculation is specified in FloatingRateCalculation. In case the reset frequency is of value T (term), the period is defined by the swap\swapStream\calculationPerioDates\effectiveDate and the swap\swapStream\calculationPerioDates\terminationDate.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <periodMultiplier> xsd:integer </periodMultiplier> [1]
   'A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative
   to another date, e.g. -2 days. If the period value is T (Term) then periodMultiplier must contain the value
   1.'

   <period> PeriodExtendedEnum </period> [1]
   'A time period, e.g. a day, week, month, year or term of the stream. If the periodMultiplier value is 0 (zero)
   then period must contain the value D (day).'

   <weeklyRollConvention> WeeklyRollConventionEnum </weeklyRollConvention> [0..1]
   'The day of the week on which a weekly reset date occurs. This element must be included if the reset
   frequency is defined as weekly and not otherwise.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ResetFrequency">
  <xsd:complexContent>
    <xsd:extension base=" Frequency ">
      <xsd:sequence>
        <xsd:element name="weeklyRollConvention" type=" WeeklyRollConventionEnum " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

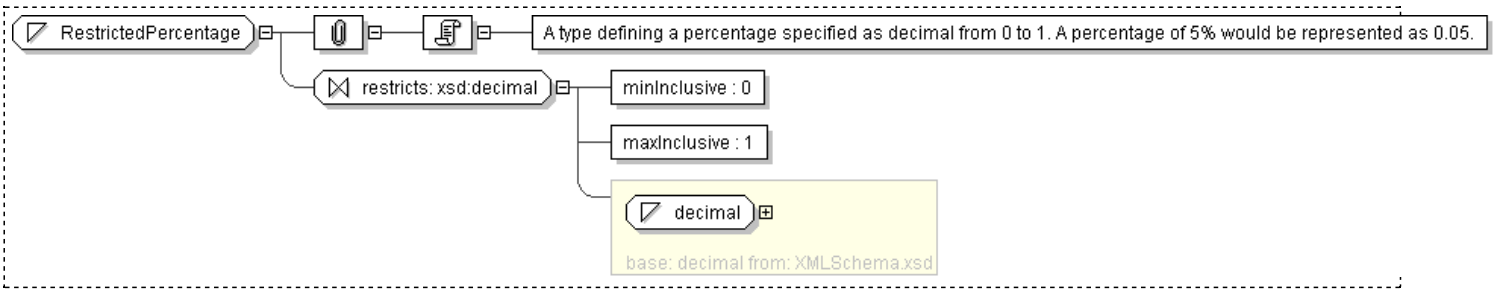
Simple Type: RestrictedPercentage

[Table of contents]

Super-types:	xsd:decimal < RestrictedPercentage (by restriction)
Sub-types:	None

Name	RestrictedPercentage
Used by (from the same schema document)	Complex Type AverageDailyTradingVolumeLimit
Content	• Base XSD Type: decimal • 0 <= *value* <= 1
Documentation	A type defining a percentage specified as decimal from 0 to 1. A percentage of 5% would be represented as 0.05.

Diagram



Schema Component Representation

```
<xsd:simpleType name="RestrictedPercentage">
  <xsd:restriction base=" xsd:decimal ">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReturnSwapNotionalAmountReference

[Table of contents]

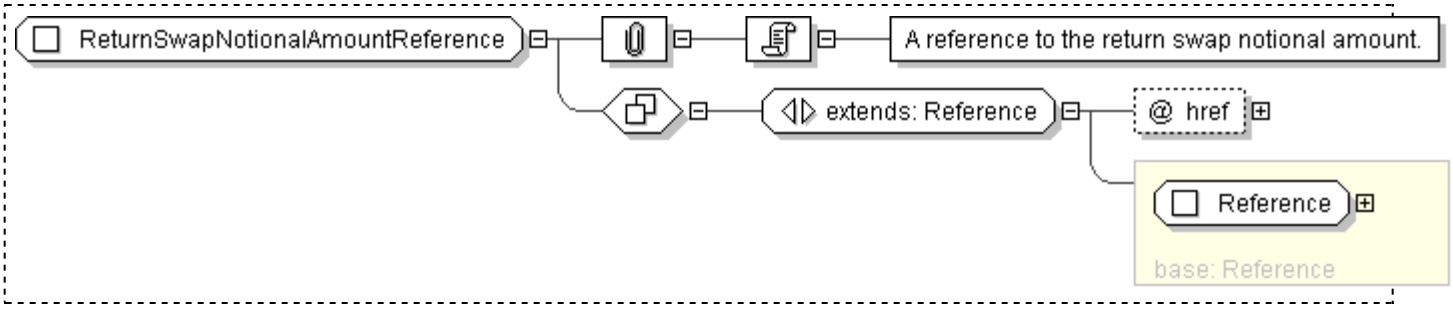
Super-types:	Reference < ReturnSwapNotionalAmountReference (by extension)
Sub-types:	None

Name	ReturnSwapNotionalAmountReference
Abstract	no
Documentation	A reference to the return swap notional amount.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapNotionalAmountReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="NotionalAmount"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Rounding

[Table of contents]

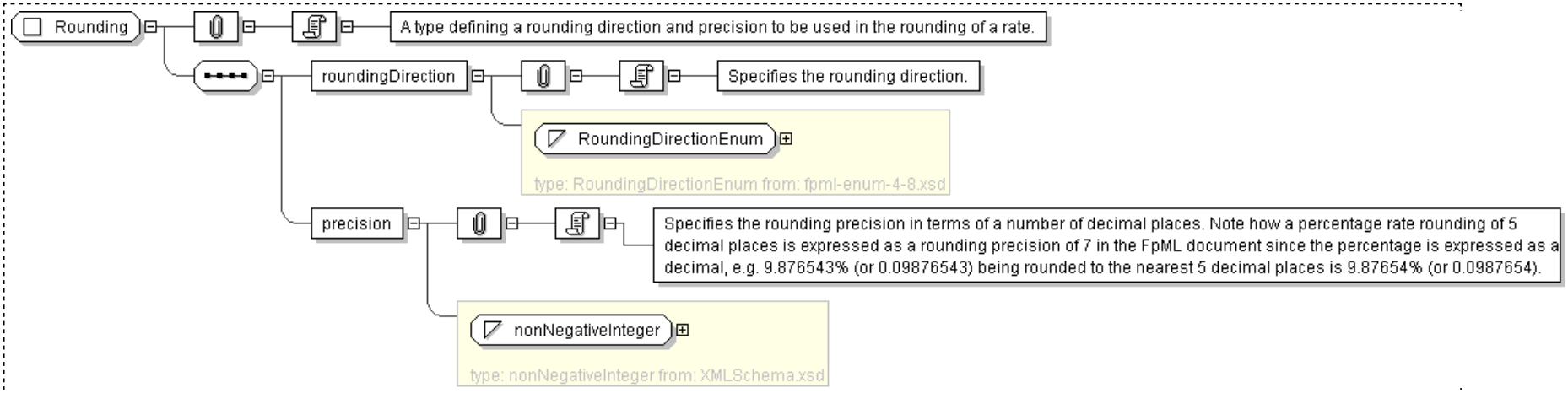
Super-types:	None
Sub-types:	None

Name	Rounding
Used by (from the same schema document)	Complex Type FloatingRateCalculation
Abstract	no
Documentation	A type defining a rounding direction and precision to be used in the rounding of a rate.

XML Instance Representation

```
<...>
   <roundingDirection> RoundingDirectionEnum </roundingDirection> [1]
    'Specifies the rounding direction.'

   <precision> xsd:nonNegativeInteger </precision> [1]
    'Specifies the rounding precision in terms of a number of decimal places. Note how a percentage rate rounding of 5 decimal
    places is expressed as a rounding precision of 7 in the FpML document since the percentage is expressed as a decimal, e.g.
    9.876543% (or 0.09876543) being rounded to the nearest 5 decimal places is 9.87654% (or 0.0987654).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Rounding">
   <xsd:sequence>
      <xsd:element name="roundingDirection" type=" RoundingDirectionEnum "/>
      <xsd:element name="precision" type=" xsd:nonNegativeInteger "/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Routing

[Table of contents]

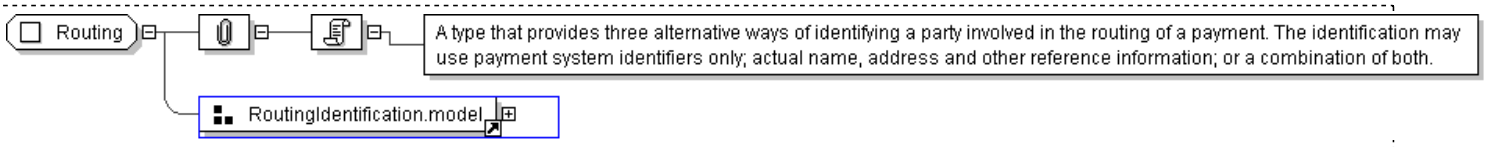
Super-types:	None
Sub-types:	None

Name	Routing
Used by (from the same schema document)	Complex Type SplitSettlement , Complex Type SplitSettlement
Abstract	no
Documentation	A type that provides three alternative ways of identifying a party involved in the routing of a payment. The identification may use payment system identifiers only; actual name, address and other reference information; or a combination of both.

XML Instance Representation

```
<...>
  Start Choice [1]
    <routingIds> RoutingIds </routingIds> [1]
     'A set of unique identifiers for a party, eachone identifying the party within a payment system. The
     assumption is that each party will not have more than one identifier within the same payment system.'

    <routingExplicitDetails> RoutingExplicitDetails </routingExplicitDetails> [1]
     'A set of details that is used to identify a party involved in the routing of a payment when the
     party does not have a code that identifies it within one of the recognized payment systems.'

    <routingIdsAndExplicitDetails> RoutingIdsAndExplicitDetails </routingIdsAndExplicitDetails> [1]
     'A combination of coded payment system identifiers and details for physical addressing for a party
     involved in the routing of a payment.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Routing">
  <xsd:group ref=" RoutingIdentification.model "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RoutingExplicitDetails

[Table of contents]

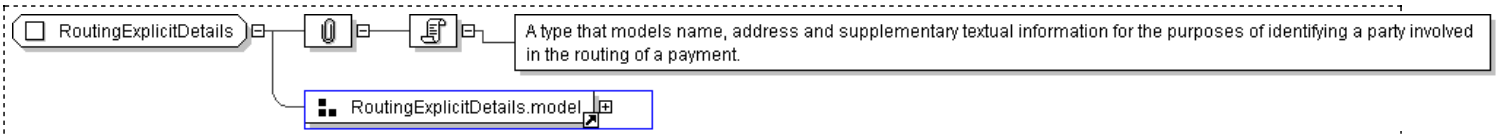
Super-types:	None
Sub-types:	None

Name	RoutingExplicitDetails
Used by (from the same schema document)	Model Group RoutingIdentification.model
Abstract	no
Documentation	A type that models name, address and supplementary textual information for the purposes of identifying a party involved in the routing of a payment.

XML Instance Representation

```
<...>
  <routingName> xsd:string </routingName> [1]
  'A real name that is used to identify a party involved in the routing of a payment.'

  <routingAddress> Address </routingAddress> [0..1]
  'A physical postal address via which a payment can be routed.'

  <routingAccountNumber> xsd:string </routingAccountNumber> [0..1]
  'An account number via which a payment can be routed.'

  <routingReferenceText> xsd:string </routingReferenceText> [0..*]
  'A piece of free-format text used to assist the identification of a party involved in the routing of a payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RoutingExplicitDetails">
   <xsd:group ref=" RoutingExplicitDetails.model "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RoutingId

[Table of contents]

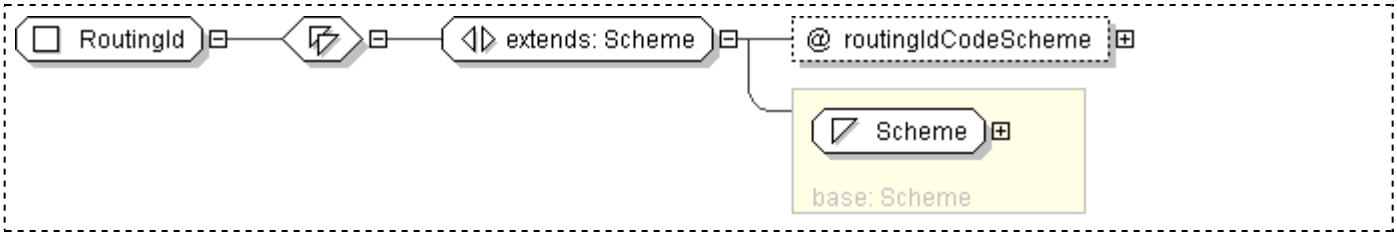
Super-types:	xsd:normalizedString < Scheme (by restriction) < RoutingId (by extension)
Sub-types:	None

Name	RoutingId
Used by (from the same schema document)	Complex Type RoutingIds
Abstract	no

XML Instance Representation

```
<...
 routingIdCodeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RoutingId">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="routingIdCodeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/ext/iso9362"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RoutingIds

[Table of contents]

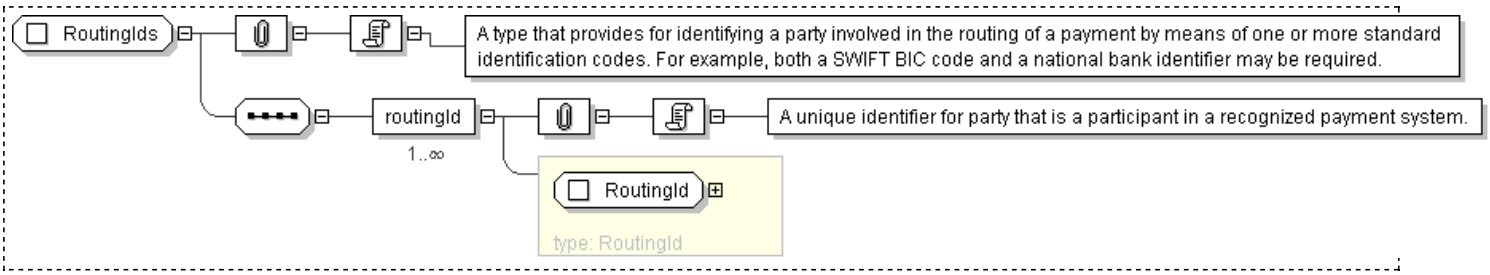
Super-types:	None
Sub-types:	None

Name	RoutingIds
Used by (from the same schema document)	Complex Type RoutingIdsAndExplicitDetails , Model Group RoutingIdentification.model
Abstract	no
Documentation	A type that provides for identifying a party involved in the routing of a payment by means of one or more standard identification codes. For example, both a SWIFT BIC code and a national bank identifier may be required.

XML Instance Representation

```
<...>
   <routingId> RoutingId </routingId> [1..*]
   'A unique identifier for party that is a participant in a recognized payment system.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RoutingIds">
   <xsd:sequence>
      <xsd:element name="routingId" type=" RoutingId " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RoutingIdsAndExplicitDetails

[Table of contents]

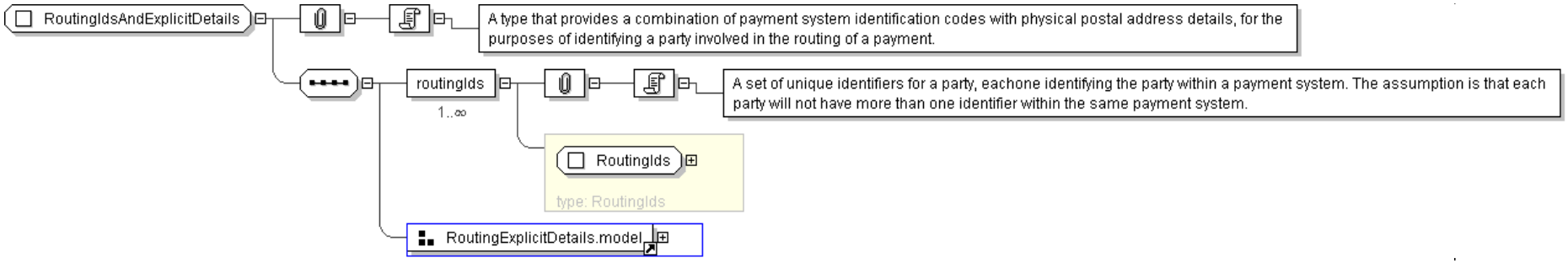
Super-types:	None
Sub-types:	None

Name	RoutingIdsAndExplicitDetails
Used by (from the same schema document)	Model Group RoutingIdentification.model
Abstract	no
Documentation	A type that provides a combination of payment system identification codes with physical postal address details, for the purposes of identifying a party involved in the routing of a payment.

XML Instance Representation

```
<...>
  <routingIds> RoutingIds </routingIds> [1..*]
  'A set of unique identifiers for a party, eachone identifying the party within a payment system. The assumption is that each party will not
  have more than one identifier within the same payment system.'

  <routingName> xsd:string </routingName> [1]
  'A real name that is used to identify a party involved in the routing of a payment.'

  <routingAddress> Address </routingAddress> [0..1]
  'A physical postal address via which a payment can be routed.'

  <routingAccountNumber> xsd:string </routingAccountNumber> [0..1]
  'An account number via which a payment can be routed.'

  <routingReferenceText> xsd:string </routingReferenceText> [0..*]
  'A piece of free-format text used to assist the identification of a party involved in the routing of a payment.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RoutingIdsAndExplicitDetails">
  <xsd:sequence>
    <xsd:element name="routingIds" type=" RoutingIds " maxOccurs="unbounded"/>
    <xsd:group ref=" RoutingExplicitDetails.model "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Schedule

[Table of contents]

Super-types: None

Sub-types:

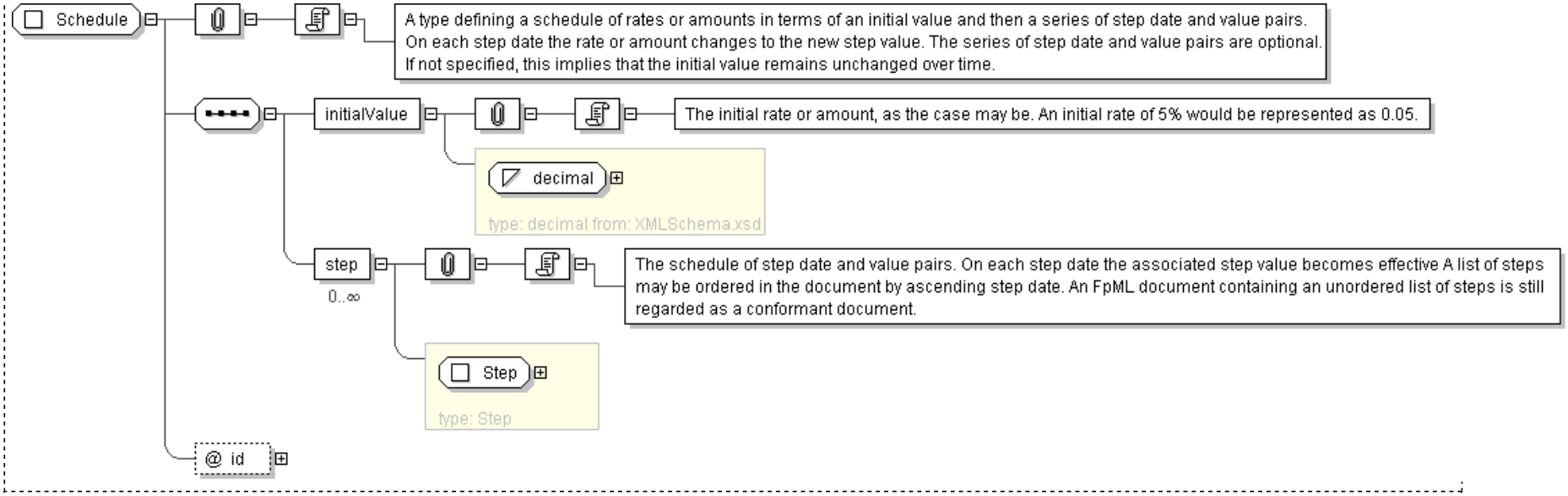
- AmountSchedule (by extension)
- SpreadSchedule (by extension)
- StrikeSchedule (by extension)

Name	Schedule
Used by (from the same schema document)	Complex Type ExerciseFeeSchedule , Complex Type FloatingRate
Abstract	no
Documentation	A type defining a schedule of rates or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> xsd:decimal </initialValue> [1]
    'The initial rate or amount, as the case may be. An initial rate of 5% would be represented as 0.05.'

   <step> Step </step> [0..*]
    'The schedule of step date and value pairs. On each step date the associated step value becomes effective A list of steps
    may be ordered in the document by ascending step date. An FpML document containing an unordered list of steps is still
    regarded as a conformant document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Schedule">
  <xsd:sequence>
    <xsd:element name="initialValue" type=" xsd:decimal "/>
    <xsd:element name="step" type=" Step " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ScheduleReference

[Table of contents]

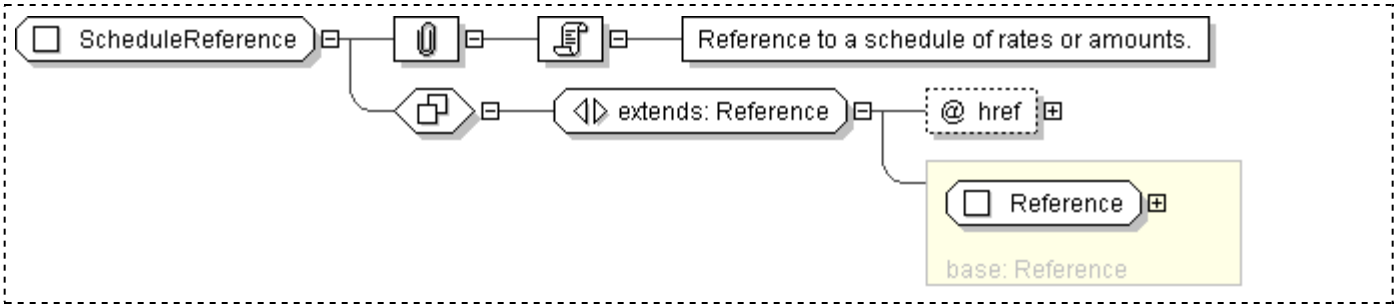
Super-types:	Reference < ScheduleReference (by extension)
Sub-types:	None

Name	ScheduleReference
Used by (from the same schema document)	Complex Type ExerciseFeeSchedule
Abstract	no
Documentation	Reference to a schedule of rates or amounts.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ScheduleReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="Schedule"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Simple Type: Scheme

[Table of contents]

Super-types: xsd:normalizedString < Scheme (by restriction)

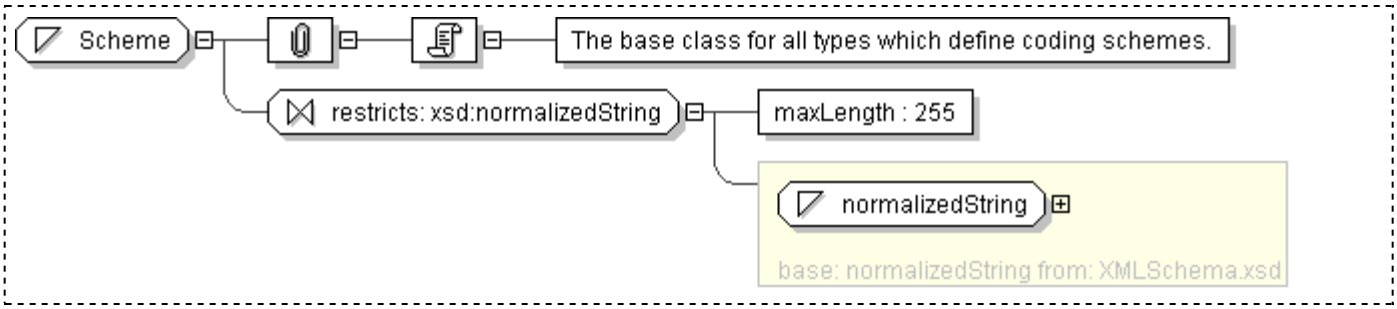
Sub-types:

- AccountId (by extension)
- BrokerConfirmationType (by extension)
- BusinessCenter (by extension)
- CashflowType (by extension)
- ClearanceSystem (by extension)
- ContractualDefinitions (by extension)
- ContractualSupplement (by extension)
- Country (by extension)
- CreditSeniority (by extension)
- CreditSupportAgreementType (by extension)
- Currency (by extension)
- DayCountFraction (by extension)
- DeterminationMethod (by extension)
- EntityId (by extension)
- EntityName (by extension)
- ExchangeId (by extension)
- FloatingRateIndex (by extension)
- GoverningLaw (by extension)
- InformationProvider (by extension)
- InstrumentId (by extension)
- InterpolationMethod (by extension)
- MainPublication (by extension)
- MasterAgreementType (by extension)
- MasterAgreementVersion (by extension)
- MasterConfirmationAnnexType (by extension)
- MasterConfirmationType (by extension)
- MatrixType (by extension)
- MatrixTerm (by extension)
- MimeType (by extension)
- PartyId (by extension)
- PaymentType (by extension)
- ProductId (by extension)
- ProductType (by extension)
- RateSourcePage (by extension)
- ReferenceAmount (by extension)
- ReferenceBankId (by extension)
- RoutingId (by extension)
- SettlementMethod (by extension)
- SettlementPriceDefaultElection (by extension)
- SettlementPriceSource (by extension)
- SpreadScheduleType (by extension)
- TimezoneLocation (by extension)

Name	Scheme
Content	• Base XSD Type: normalizedString • *length* <= 255

Documentation	The base class for all types which define coding schemes.

Diagram



Schema Component Representation

```
<xsd:simpleType name="Scheme">
   <xsd:restriction base=" xsd:normalizedString ">
      <xsd:maxLength value="255"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementInformation

[Table of contents]

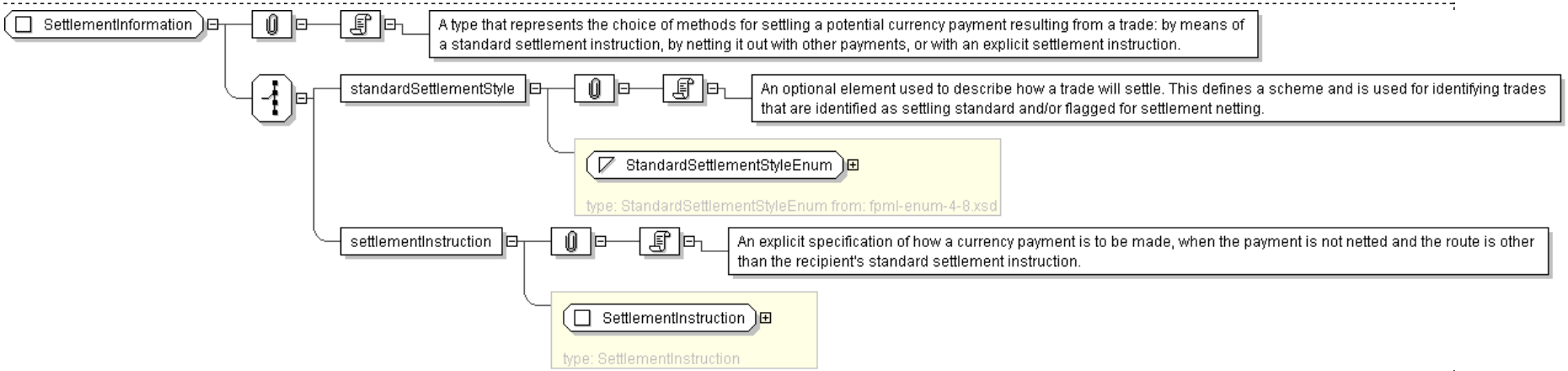
Super-types:	None
Sub-types:	None

Name	SettlementInformation
Used by (from the same schema document)	Complex Type Payment
Abstract	no
Documentation	A type that represents the choice of methods for settling a potential currency payment resulting from a trade: by means of a standard settlement instruction, by netting it out with other payments, or with an explicit settlement instruction.

XML Instance Representation

```
<...>
  Start Choice [1]
    <standardSettlementStyle> StandardSettlementStyleEnum </standardSettlementStyle> [1]
      'An optional element used to describe how a trade will settle. This defines a scheme and is used for identifying trades that are
      identified as settling standard and/or flagged for settlement netting.'

    <settlementInstruction> SettlementInstruction </settlementInstruction> [1]
      'An explicit specification of how a currency payment is to be made, when the payment is not netted and the route is other than the
      recipient\'s standard settlement instruction.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementInformation">
  <xsd:choice>
    <xsd:element name="standardSettlementStyle" type=" StandardSettlementStyleEnum "/>
    <xsd:element name="settlementInstruction" type=" SettlementInstruction "/>
  </xsd:choice>
</xsd:complexType>
```

XML Schema Documentation

Complex Type: SettlementInstruction

[Table of contents]

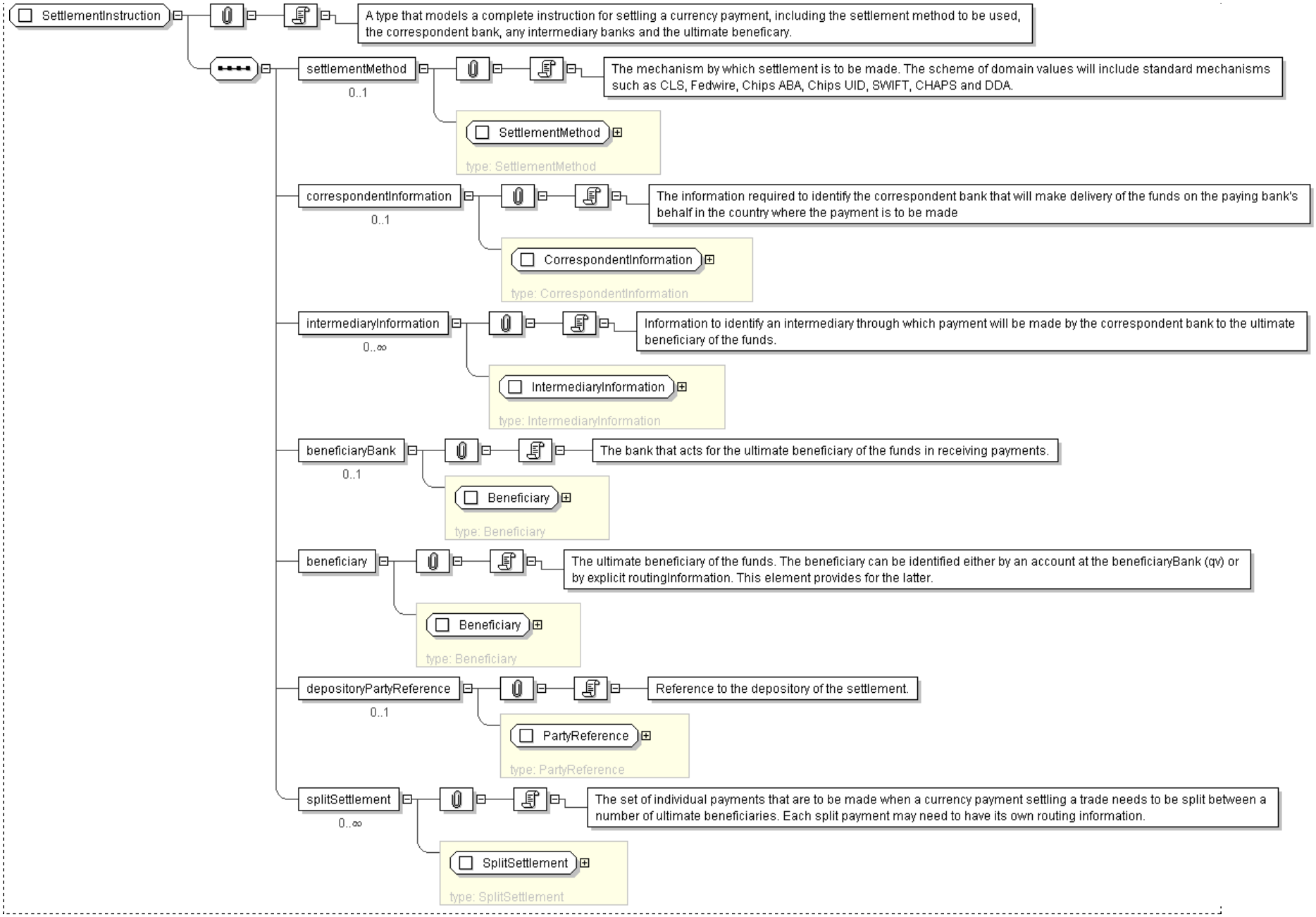
Super-types:	None
Sub-types:	None

Name	SettlementInstruction
Used by (from the same schema document)	Complex Type SettlementInformation
Abstract	no
Documentation	A type that models a complete instruction for settling a currency payment, including the settlement method to be used, the correspondent bank, any intermediary banks and the ultimate beneficary.

XML Instance Representation

```
<...>
   <settlementMethod> SettlementMethod </settlementMethod> [0..1]
   'The mechanism by which settlement is to be made. The scheme of domain values will include standard mechanisms such as CLS, Fedwire, Chips
   ABA, Chips UID, SWIFT, CHAPS and DDA.'

   <correspondentInformation> CorrespondentInformation </correspondentInformation> [0..1]
   'The information required to identify the correspondent bank that will make delivery of the funds on the paying bank\'s behalf in the country
   where the payment is to be made'

   <intermediaryInformation> IntermediaryInformation </intermediaryInformation> [0..*]
   'Information to identify an intermediary through which payment will be made by the correspondent bank to the ultimate beneficiary of the
   funds.'

   <beneficiaryBank> Beneficiary </beneficiaryBank> [0..1]
   'The bank that acts for the ultimate beneficiary of the funds in receiving payments.'

   <beneficiary> Beneficiary </beneficiary> [1]
   'The ultimate beneficiary of the funds. The beneficiary can be identified either by an account at the beneficiaryBank (qv) or by explicit
   routingInformation. This element provides for the latter.'

   <depositoryPartyReference> PartyReference </depositoryPartyReference> [0..1]
   'Reference to the depository of the settlement.'

   <splitSettlement> SplitSettlement </splitSettlement> [0..*]
   'The set of individual payments that are to be made when a currency payment settling a trade needs to be split between a number of ultimate
   beneficiaries. Each split payment may need to have its own routing information.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementInstruction">
  <xsd:sequence>
    <xsd:element name="settlementMethod" type=" SettlementMethod " minOccurs="0"/>
    <xsd:element name="correspondentInformation" type=" CorrespondentInformation " minOccurs="0"/>
    <xsd:element name="intermediaryInformation" type=" IntermediaryInformation " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="beneficiaryBank" type=" Beneficiary " minOccurs="0"/>
    <xsd:element name="beneficiary" type=" Beneficiary "/>
    <xsd:element name="depositoryPartyReference" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="splitSettlement" type=" SplitSettlement " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementMethod

[Table of contents]

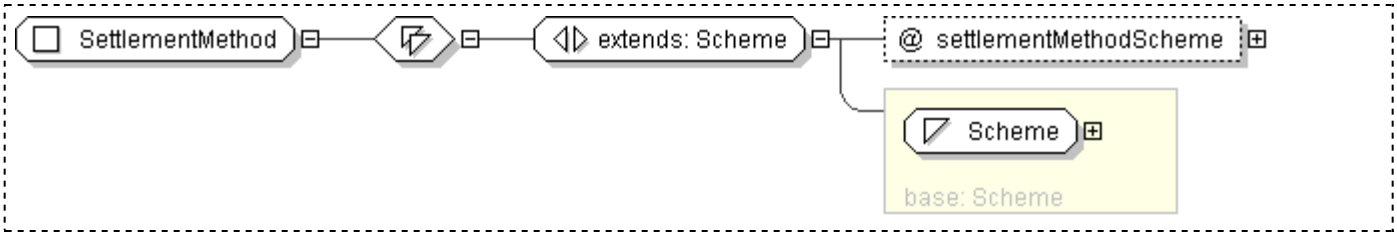
Super-types:	xsd:normalizedString < Scheme (by restriction) < SettlementMethod (by extension)
Sub-types:	None

Name	SettlementMethod
Used by (from the same schema document)	Complex Type SettlementInstruction
Abstract	no

XML Instance Representation

```
<...
 settlementMethodScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementMethod">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="settlementMethodScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/settlement-method"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPriceDefaultElection

[Table of contents]

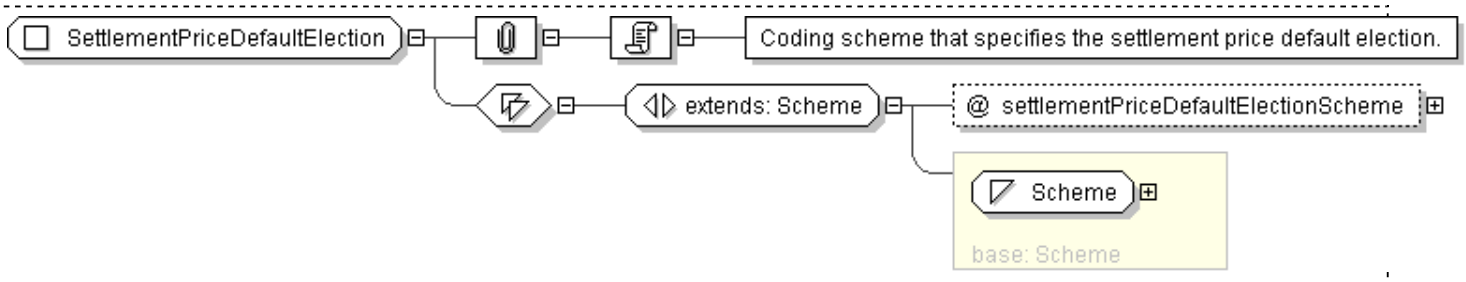
Super-types:	xsd:normalizedString < Scheme (by restriction) < SettlementPriceDefaultElection (by extension)
Sub-types:	None

Name	SettlementPriceDefaultElection
Abstract	no
Documentation	Coding scheme that specifies the settlement price default election.

XML Instance Representation

```
<...
 settlementPriceDefaultElectionScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPriceDefaultElection">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="settlementPriceDefaultElectionScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/settlement-price-default-election"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementPriceSource

[Table of contents]

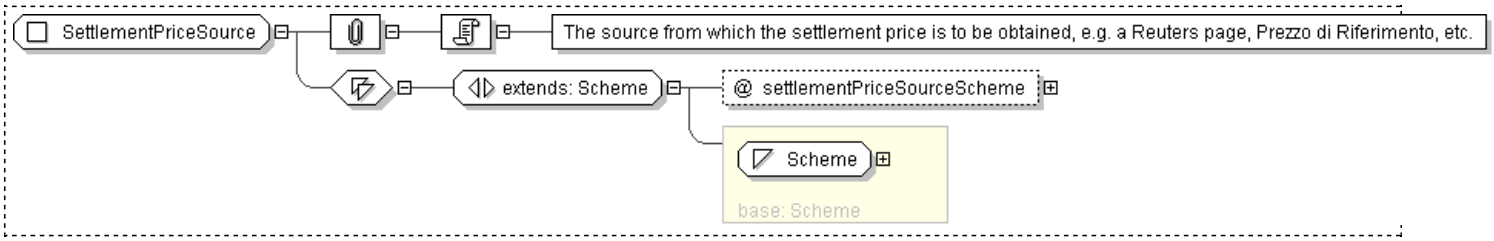
Super-types:	xsd:normalizedString < Scheme (by restriction) < SettlementPriceSource (by extension)
Sub-types:	None

Name	SettlementPriceSource
Abstract	no
Documentation	The source from which the settlement price is to be obtained, e.g. a Reuters page, Prezzo di Riferimento, etc.

XML Instance Representation

```
<...
 settlementPriceSourceScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementPriceSource">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="settlementPriceSourceScheme" type=" xsd:anyURI " default="http://www.fpml.org/coding-
      scheme/settlement-price-source"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SettlementRateSource

[Table of contents]

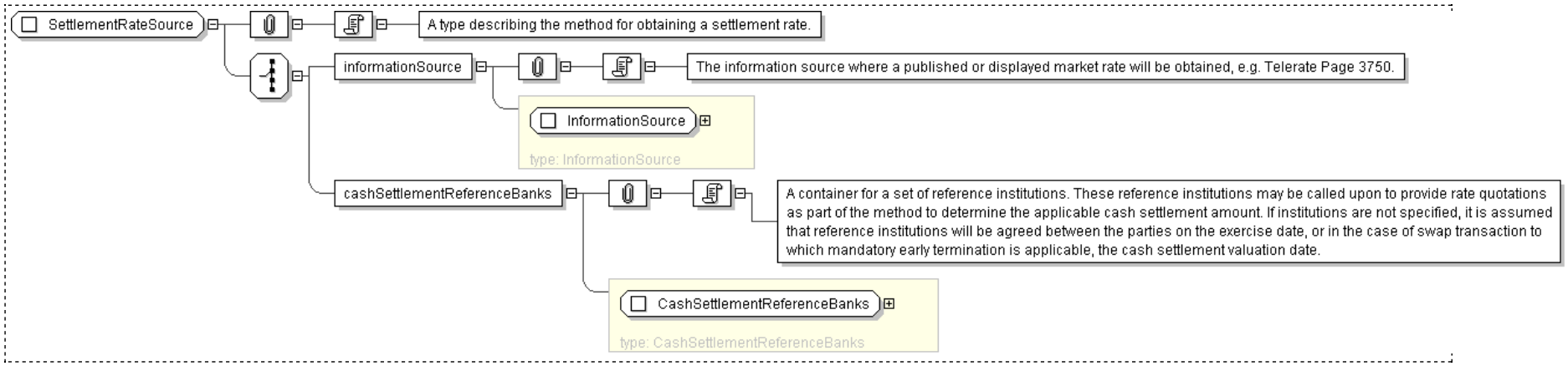
Super-types:	None
Sub-types:	None

Name	SettlementRateSource
Abstract	no
Documentation	A type describing the method for obtaining a settlement rate.

XML Instance Representation

```
<...>
  Start Choice [1]
    <informationSource> InformationSource </informationSource> [1]
    'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.'

    <cashSettlementReferenceBanks> CashSettlementReferenceBanks </cashSettlementReferenceBanks> [1]
    'A container for a set of reference institutions. These reference institutions may be called upon to provide rate quotations as part of the
    method to determine the applicable cash settlement amount. If institutions are not specified, it is assumed that reference institutions will be
    agreed between the parties on the exercise date, or in the case of swap transaction to which mandatory early termination is applicable, the cash
    settlement valuation date.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SettlementRateSource">
  <xsd:choice>
    <xsd:element name="informationSource" type=" InformationSource "/>
    <xsd:element name="cashSettlementReferenceBanks" type=" CashSettlementReferenceBanks "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SharedAmericanExercise

[Table of contents]

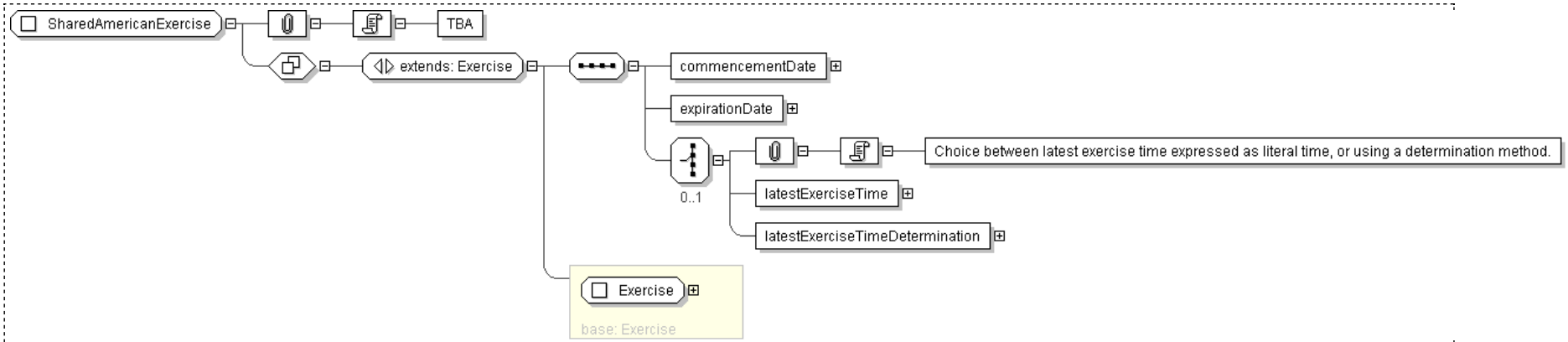
Super-types:	Exercise < SharedAmericanExercise (by extension)
Sub-types:	None

Name	SharedAmericanExercise
Abstract	no
Documentation	TBA

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <commencementDate> AdjustableOrRelativeDate </commencementDate> [1]
    'The first day of the exercise period for an American style option.'

   <expirationDate> AdjustableOrRelativeDate </expirationDate> [1]
    'The last day within an exercise period for an American style option. For a European style option it is the only day within the exercise period.'

   Start Choice [0..1]
'Choice between latest exercise time expressed as literal time, or using a determination method.'


      <latestExerciseTime> BusinessCenterTime </latestExerciseTime> [1]
       'For a Bermuda or American style option, the latest time on an exercise business day (excluding the expiration date) within the exercise period
       that notice can be given by the buyer to the seller or seller\'s agent. Notice of exercise given after this time will be deemed to have been
       given on the next exercise business day.'

      <latestExerciseTimeDetermination> DeterminationMethod </latestExerciseTimeDetermination> [1]
       'Latest exercise time determination method.'

   End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SharedAmericanExercise">
  <xsd:complexContent>
    <xsd:extension base=" Exercise ">
      <xsd:sequence>
        <xsd:element name="commencementDate" type=" AdjustableOrRelativeDate "/>
        <xsd:element name="expirationDate" type=" AdjustableOrRelativeDate "/>
        <xsd:choice minOccurs="0">
```

```
            <xsd:element name="latestExerciseTime" type=" BusinessCenterTime "/>
            <xsd:element name="latestExerciseTimeDetermination" type=" DeterminationMethod "/>
          </xsd:choice>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SimplePayment

[Table of contents]

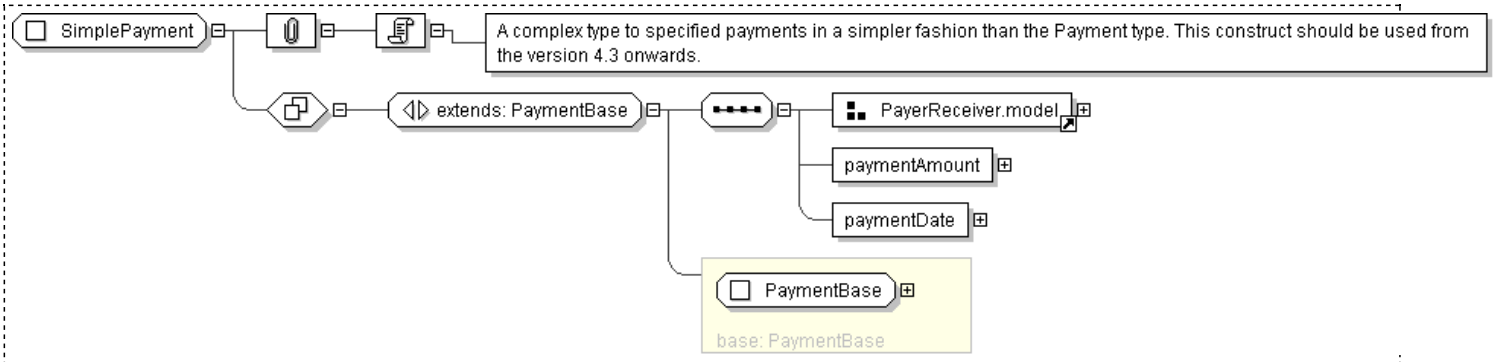
Super-types:	PaymentBase < SimplePayment (by extension)
Sub-types:	None

Name	SimplePayment
Abstract	no
Documentation	A complex type to specified payments in a simpler fashion than the Payment type. This construct should be used from the version 4.3 onwards.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentAmount> Money </paymentAmount> [1]
   <paymentDate> AdjustableOrRelativeAndAdjustedDate </paymentDate> [1]
   'The payment date. This date is subject to adjustment in accordance with any applicable business day
   convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SimplePayment">
   <xsd:complexContent>
      <xsd:extension base=" PaymentBase ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="paymentAmount" type=" Money "/>
            <xsd:element name="paymentDate" type=" AdjustableOrRelativeAndAdjustedDate "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SplitSettlement

[Table of contents]

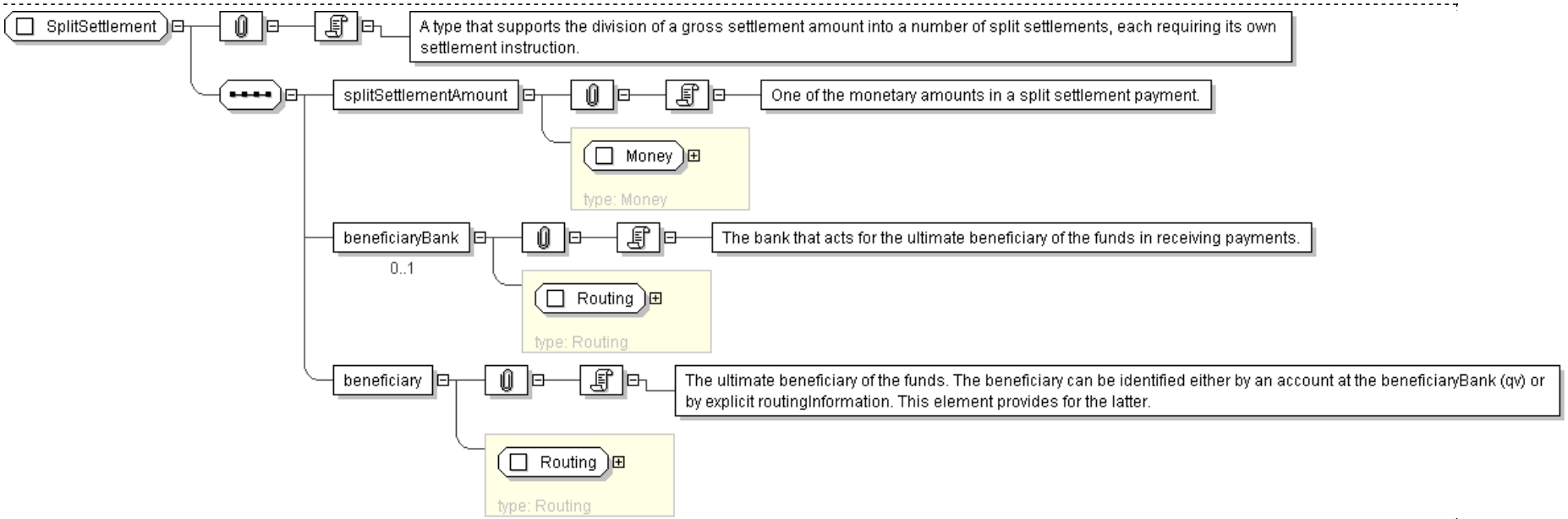
Super-types:	None
Sub-types:	None

Name	SplitSettlement
Used by (from the same schema document)	Complex Type SettlementInstruction
Abstract	no
Documentation	A type that supports the division of a gross settlement amount into a number of split settlements, each requiring its own settlement instruction.

XML Instance Representation

```
<...>
   <splitSettlementAmount> Money </splitSettlementAmount> [1]
   'One of the monetary amounts in a split settlement payment.'

   <beneficiaryBank> Routing </beneficiaryBank> [0..1]
   'The bank that acts for the ultimate beneficiary of the funds in receiving payments.'

   <beneficiary> Routing </beneficiary> [1]
   'The ultimate beneficiary of the funds. The beneficiary can be identified either by an account at the beneficiaryBank (qv) or by
   explicit routingInformation. This element provides for the latter.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SplitSettlement">
  <xsd:sequence>
    <xsd:element name="splitSettlementAmount" type=" Money "/>
    <xsd:element name="beneficiaryBank" type=" Routing " minOccurs="0"/>
    <xsd:element name="beneficiary" type=" Routing "/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SpreadSchedule

[Table of contents]

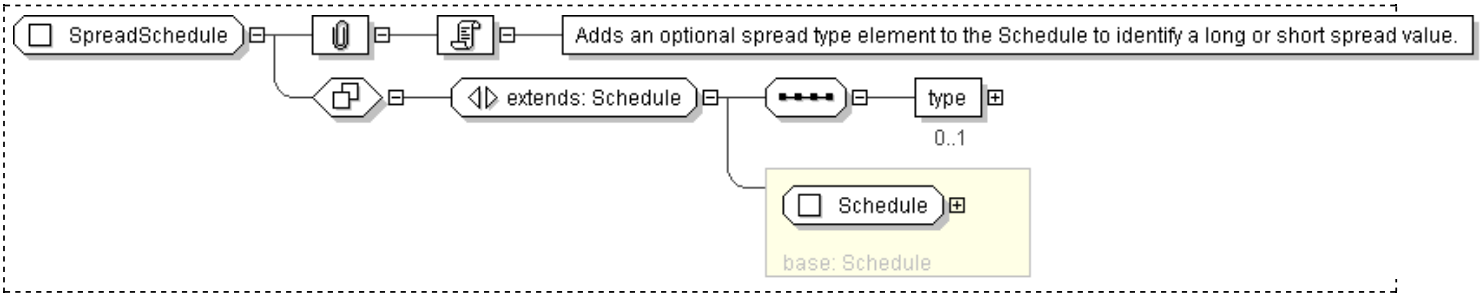
Super-types:	Schedule < SpreadSchedule (by extension)
Sub-types:	None

Name	SpreadSchedule
Used by (from the same schema document)	Complex Type FloatingRate
Abstract	no
Documentation	Adds an optional spread type element to the Schedule to identify a long or short spread value.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> xsd:decimal </initialValue> [1]
   'The initial rate or amount, as the case may be. An initial rate of 5% would be represented
   as 0.05.'

   <step> Step </step> [0..*]
   'The schedule of step date and value pairs. On each step date the associated step value
   becomes effective A list of steps may be ordered in the document by ascending step date. An
   FpML document containing an unordered list of steps is still regarded as a conformant
   document.'

   <type> SpreadScheduleType </type> [0..1]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SpreadSchedule">
   <xsd:complexContent>
      <xsd:extension base=" Schedule ">
         <xsd:sequence>
            <xsd:element name="type" type=" SpreadScheduleType " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SpreadScheduleReference

[Table of contents]

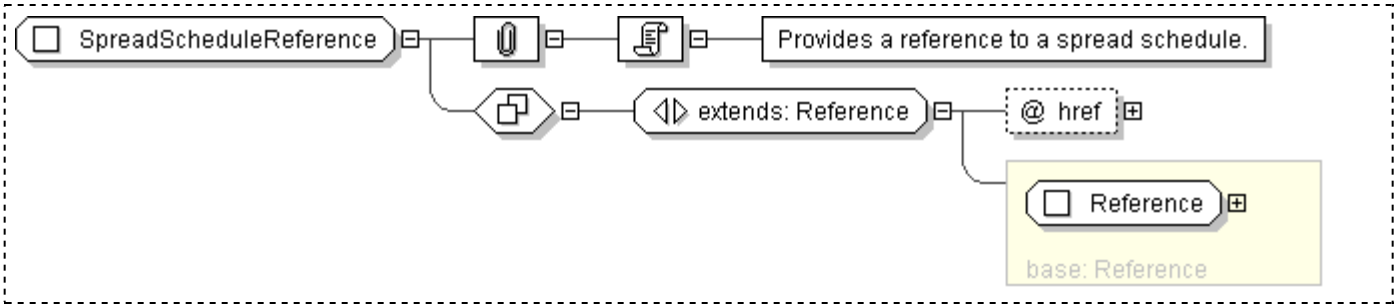
Super-types:	Reference < SpreadScheduleReference (by extension)
Sub-types:	None

Name	SpreadScheduleReference
Abstract	no
Documentation	Provides a reference to a spread schedule.

XML Instance Representation

```
<...
 href=" xsd:IDREF [1]"/>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SpreadScheduleReference">
   <xsd:complexContent>
      <xsd:extension base=" Reference ">
         <xsd:attribute name="href" type=" xsd:IDREF " use="required"
         reference="SpreadSchedule"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SpreadScheduleType

[Table of contents]

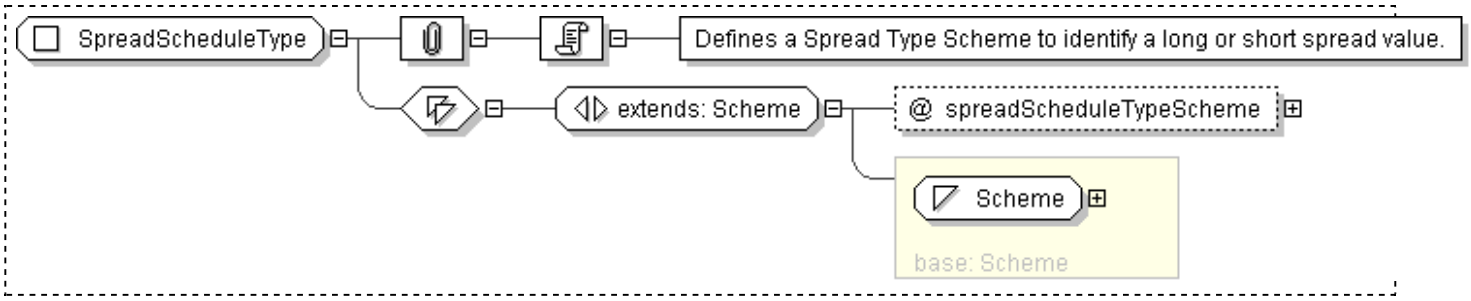
Super-types:	xsd:normalizedString < Scheme (by restriction) < SpreadScheduleType (by extension)
Sub-types:	None

Name	SpreadScheduleType
Used by (from the same schema document)	Complex Type SpreadSchedule
Abstract	no
Documentation	Defines a Spread Type Scheme to identify a long or short spread value.

XML Instance Representation

```
<...
 spreadScheduleTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SpreadScheduleType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="spreadScheduleTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/spread-schedule-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Step

[Table of contents]

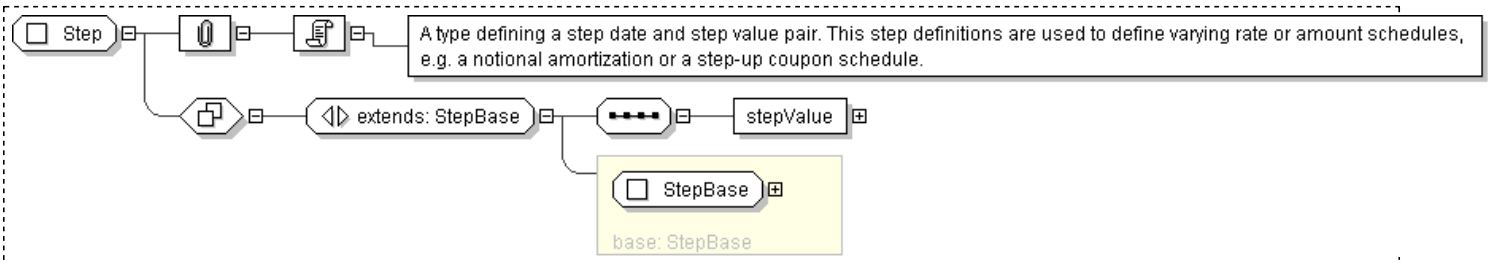
Super-types:	StepBase < Step (by extension)
Sub-types:	None

Name	Step
Used by (from the same schema document)	Complex Type Schedule
Abstract	no
Documentation	A type defining a step date and step value pair. This step definitions are used to define varying rate or amount schedules, e.g. a notional amortization or a step-up coupon schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <stepDate> xsd:date </stepDate> [1]
    'The date on which the associated stepValue becomes effective. This day may be subject to adjustment
    in accordance with a business day convention.'

    <stepValue> xsd:decimal </stepValue> [1]
    'The rate or amount which becomes effective on the associated stepDate. A rate of 5% would be
    represented as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Step">
  <xsd:complexContent>
    <xsd:extension base=" StepBase ">
      <xsd:sequence>
        <xsd:element name="stepValue" type=" xsd:decimal "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StepBase

[Table of contents]

Super-types: None

Sub-types:

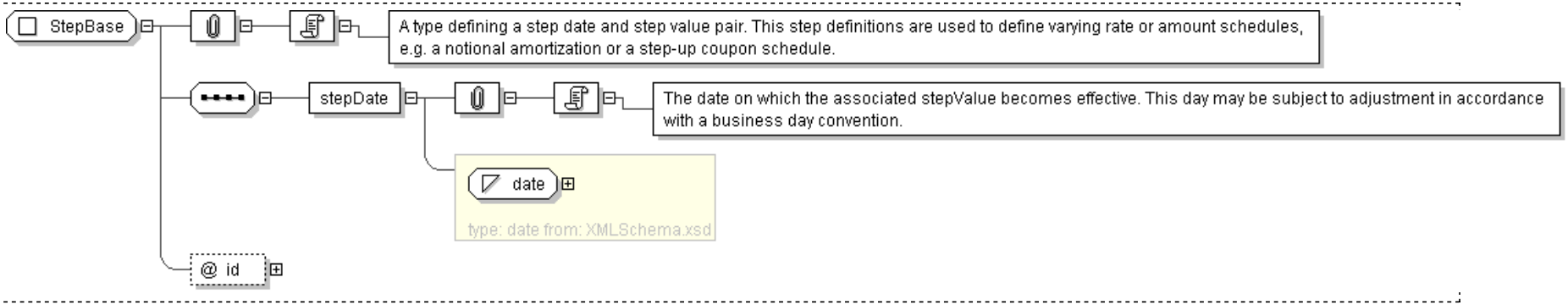
- NonNegativeStep (by extension)
- PositiveStep (by extension)
- Step (by extension)

Name	StepBase
Abstract	yes
Documentation	A type defining a step date and step value pair. This step definitions are used to define varying rate or amount schedules, e.g. a notional amortization or a step-up coupon schedule.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <stepDate> xsd:date </stepDate> [1]
    'The date on which the associated stepValue becomes effective. This day may be subject to adjustment in accordance with a
    business day convention.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StepBase" abstract="true">
  <xsd:sequence>
    <xsd:element name="stepDate" type=" xsd:date "/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StreetAddress

[Table of contents]

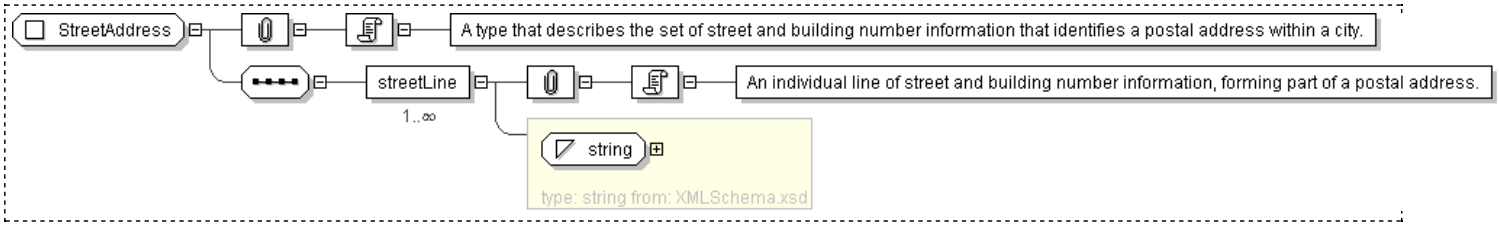
Super-types:	None
Sub-types:	None

Name	StreetAddress
Used by (from the same schema document)	Complex Type Address
Abstract	no
Documentation	A type that describes the set of street and building number information that identifies a postal address within a city.

XML Instance Representation

```
<...>
  <streetLine> xsd:string </streetLine> [1..*]
  'An individual line of street and building number information, forming part of a postal address.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StreetAddress">
  <xsd:sequence>
    <xsd:element name="streetLine" type=" xsd:string " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Strike

[Table of contents]

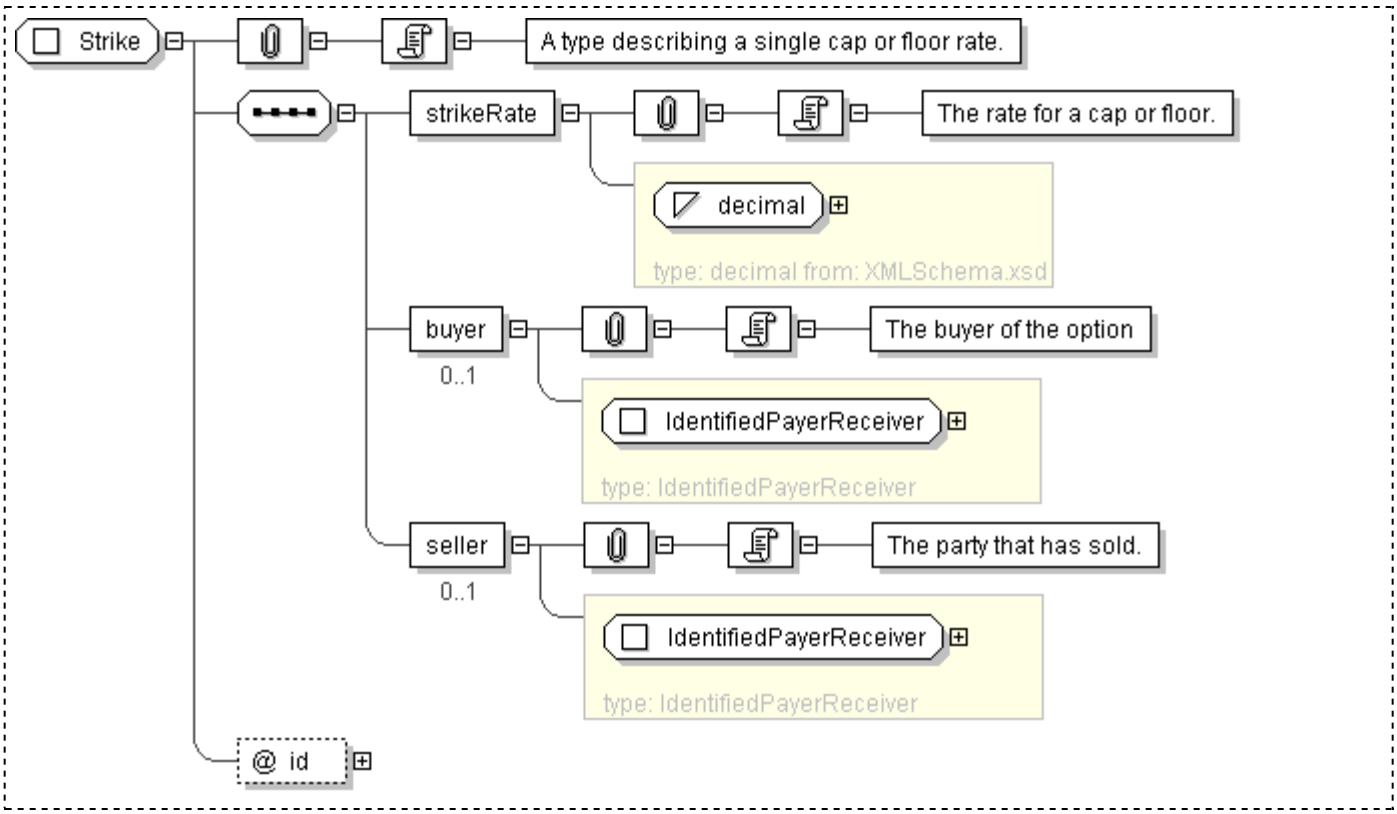
Super-types:	None
Sub-types:	None

Name	Strike
Abstract	no
Documentation	A type describing a single cap or floor rate.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <strikeRate> xsd:decimal </strikeRate> [1]
    'The rate for a cap or floor.'

   <buyer> IdentifiedPayerReceiver </buyer> [0..1]
    'The buyer of the option'

   <seller> IdentifiedPayerReceiver </seller> [0..1]
    'The party that has sold.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="      ">
```

```
                        Strike
   <xsd:sequence>
      <xsd:element name="strikeRate" type=" xsd:decimal "/>
      <xsd:element name="buyer" type=" IdentifiedPayerReceiver " minOccurs="0"/>
      <xsd:element name="seller" type=" IdentifiedPayerReceiver " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StrikeSchedule

[Table of contents]

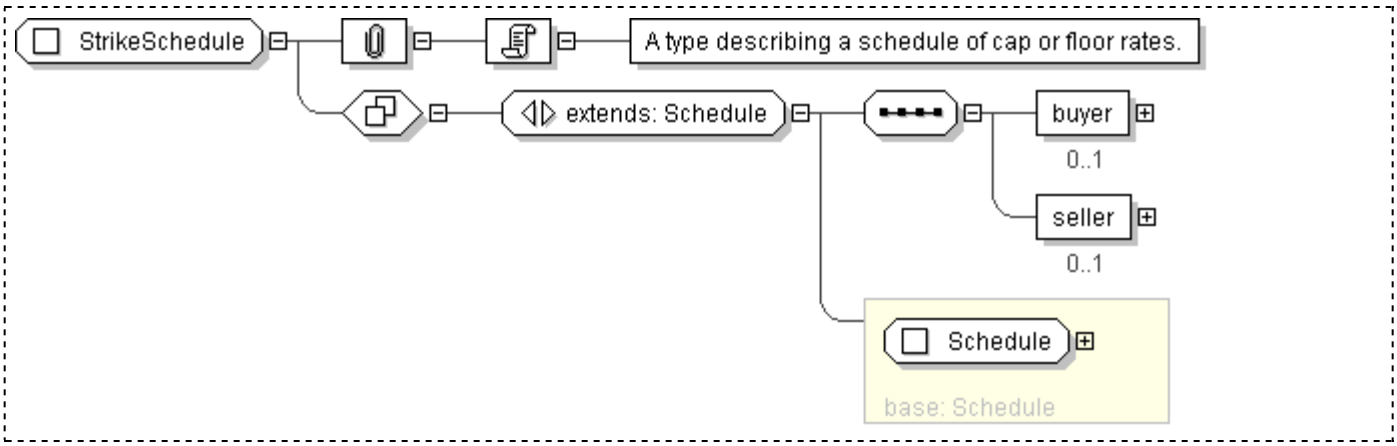
Super-types:	Schedule < StrikeSchedule (by extension)
Sub-types:	None

Name	StrikeSchedule
Used by (from the same schema document)	Complex Type FloatingRate , Complex Type FloatingRate
Abstract	no
Documentation	A type describing a schedule of cap or floor rates.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <initialValue> xsd:decimal </initialValue> [1]
    'The initial rate or amount, as the case may be. An initial rate of 5% would be
    represented as 0.05.'

   <step> Step </step> [0..*]
    'The schedule of step date and value pairs. On each step date the associated
    step value becomes effective A list of steps may be ordered in the document by
    ascending step date. An FpML document containing an unordered list of steps is
    still regarded as a conformant document.'

   <buyer> IdentifiedPayerReceiver </buyer> [0..1]
    'The buyer of the option'

   <seller> IdentifiedPayerReceiver </seller> [0..1]
    'The party that has sold.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StrikeSchedule">
   <xsd:complexContent>
      <xsd:extension base=" Schedule ">
         <xsd:sequence>
            <xsd:element name="buyer" type=" IdentifiedPayerReceiver " minOccurs="0"/>
```

```
          <xsd:element name="seller" type=" IdentifiedPayerReceiver " minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Stub

[Table of contents]

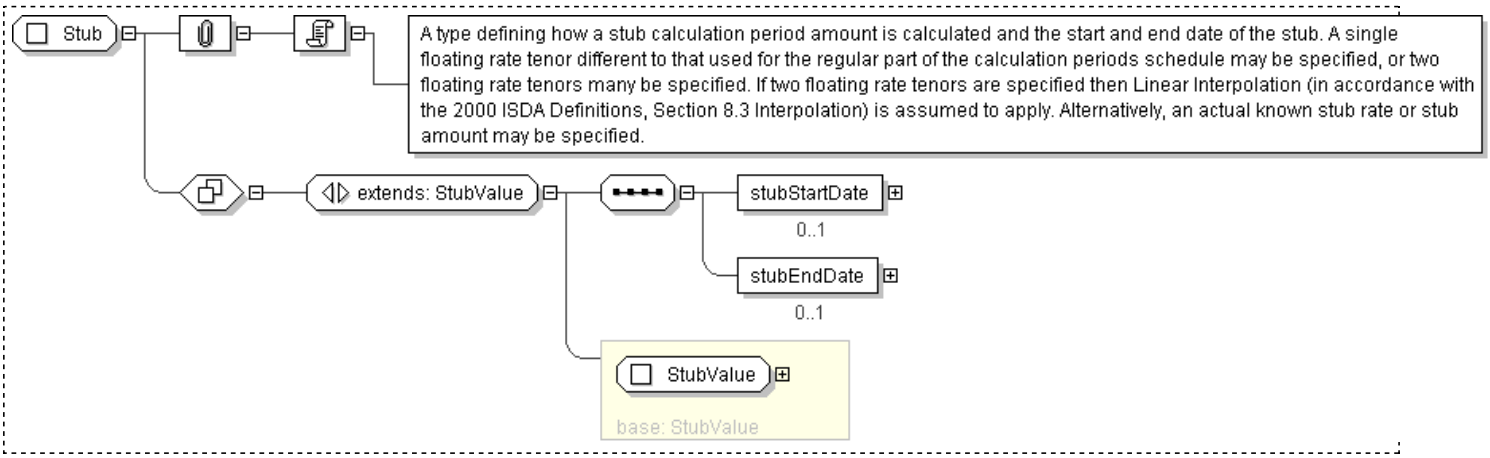
Super-types:	StubValue < Stub (by extension)
Sub-types:	None

Name	Stub
Abstract	no
Documentation	A type defining how a stub calculation period amount is calculated and the start and end date of the stub. A single floating rate tenor different to that used for the regular part of the calculation periods schedule may be specified, or two floating rate tenors many be specified. If two floating rate tenors are specified then Linear Interpolation (in accordance with the 2000 ISDA Definitions, Section 8.3 Interpolation) is assumed to apply. Alternatively, an actual known stub rate or stub amount may be specified.

XML Instance Representation

```
<...>
  Start Choice [1]
    <floatingRate> FloatingRate </floatingRate> [1..2]
      'The rates to be applied to the initial or final stub may be the linear interpolation of two
      different rates. While the majority of the time, the rate indices will be the same as that specified
      in the stream and only the tenor itself will be different, it is possible to specift two different
      rates. For example, a 2 month stub period may use the linear interpolation of a 1 month and 3 month
      rate. The different rates would be specified in this component. Note that a maximum of two rates
      can be specified. If a stub period uses the same floating rate index, including tenor, as the
      regular calculation periods then this should not be specified again within this component, i.e. the
      stub calculation period amount component may not need to be specified even if there is an initial or
      final stub period. If a stub period uses a different floating rate index compared to the regular
      calculation periods then this should be specified within this component. If specified here, they are
      likely to have id attributes, allowing them to be referenced from within the cashflows component.'

    <stubRate> xsd:decimal </stubRate> [1]
      'An actual rate to apply for the initial or final stub period may have been agreed between the
      principal parties (in a similar way to how an initial rate may have been agreed for the first
      regular period). If an actual stub rate has been agreed then it would be included in this component.
      It will be a per annum rate, expressed as a decimal. A stub rate of 5% would be represented as
      0.05.'

    <stubAmount> Money </stubAmount> [1]
      'An actual amount to apply for the initial or final stub period may have been agreed between th two
      parties. If an actual stub amount has been agreed then it would be included in this component.'

  End Choice
  <stubStartDate> AdjustableOrRelativeDate </stubStartDate> [0..1]
    'Start date of stub period. This was created to support use of the InterestRateStream within the Equity
    Derivative sphere, and this element is not expected to be produced in the representation of Interest
    Rate products.'

  <stubEndDate> AdjustableOrRelativeDate </stubEndDate> [0..1]
    'End date of stub period. This was created to support use of the InterestRateStream within the Equity
    Derivative sphere, and this element is not expected to be produced in the representation of Interest
    Rate products.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Stub">
  <xsd:complexContent>
    <xsd:extension base=" StubValue ">
      <xsd:sequence>
        <xsd:element name="stubStartDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
        <xsd:element name="stubEndDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: StubValue

[Table of contents]

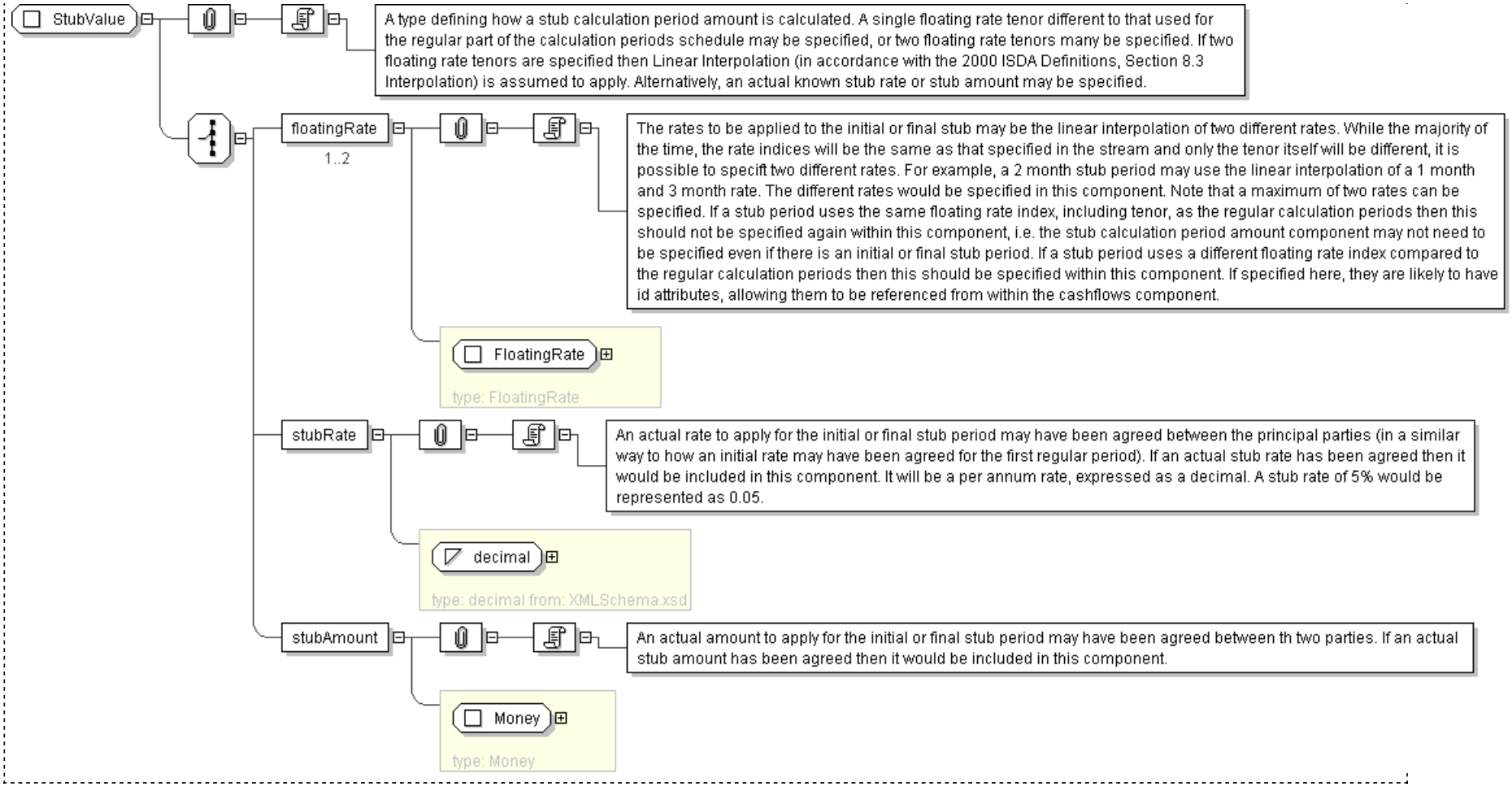
Super-types:	None
Sub-types:	• Stub (by extension)

Name	StubValue
Abstract	no
Documentation	A type defining how a stub calculation period amount is calculated. A single floating rate tenor different to that used for the regular part of the calculation periods schedule may be specified, or two floating rate tenors many be specified. If two floating rate tenors are specified then Linear Interpolation (in accordance with the 2000 ISDA Definitions, Section 8.3 Interpolation) is assumed to apply. Alternatively, an actual known stub rate or stub amount may be specified.

XML Instance Representation

```
<...>
  Start Choice [1]
    <floatingRate> FloatingRate </floatingRate> [1..2]
    'The rates to be applied to the initial or final stub may be the linear interpolation of two different rates. While the
    majority of the time, the rate indices will be the same as that specified in the stream and only the tenor itself will be
    different, it is possible to specift two different rates. For example, a 2 month stub period may use the linear interpolation
    of a 1 month and 3 month rate. The different rates would be specified in this component. Note that a maximum of two rates can
    be specified. If a stub period uses the same floating rate index, including tenor, as the regular calculation periods then
    this should not be specified again within this component, i.e. the stub calculation period amount component may not need to
    be specified even if there is an initial or final stub period. If a stub period uses a different floating rate index compared
    to the regular calculation periods then this should be specified within this component. If specified here, they are likely to
    have id attributes, allowing them to be referenced from within the cashflows component.'

    <stubRate> xsd:decimal </stubRate> [1]
    'An actual rate to apply for the initial or final stub period may have been agreed between the principal parties (in a
    similar way to how an initial rate may have been agreed for the first regular period). If an actual stub rate has been agreed
    then it would be included in this component. It will be a per annum rate, expressed as a decimal. A stub rate of 5% would be
    represented as 0.05.'

    <stubAmount> Money </stubAmount> [1]
    'An actual amount to apply for the initial or final stub period may have been agreed between th two parties. If an actual
    stub amount has been agreed then it would be included in this component.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StubValue">
  <xsd:choice>
    <xsd:element name="floatingRate" type=" FloatingRate " maxOccurs="2"/>
    <xsd:element name="stubRate" type=" xsd:decimal "/>
    <xsd:element name="stubAmount" type=" Money "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TimezoneLocation

[Table of contents]

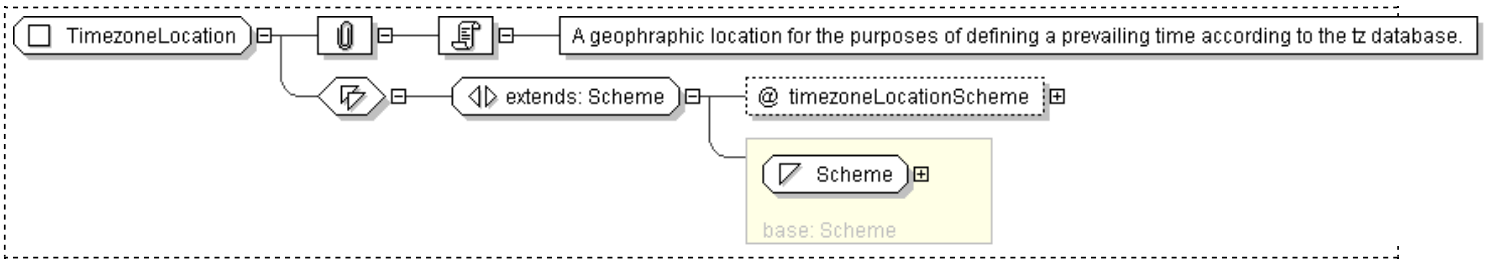
Super-types:	xsd:normalizedString < Scheme (by restriction) < TimezoneLocation (by extension)
Sub-types:	None

Name	TimezoneLocation
Used by (from the same schema document)	Complex Type PrevailingTime
Abstract	no
Documentation	A geophraphic location for the purposes of defining a prevailing time according to the tz database.

XML Instance Representation

```
<...
 timezoneLocationScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TimezoneLocation">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="timezoneLocationScheme" type=" xsd:anyURI "
      default="http://www.fpml.org/coding-scheme/external/tzdatabase"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

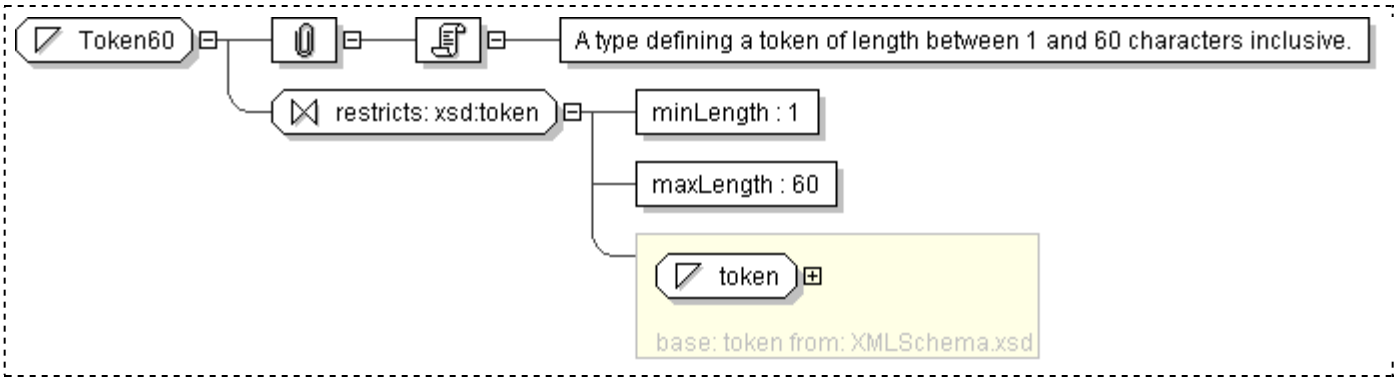
Simple Type: Token60

[Table of contents]

Super-types:	xsd:token < Token60 (by restriction)
Sub-types:	None

Name	Token60
Content	• Base XSD Type: token • *length* >= 1
Documentation	A type defining a token of length between 1 and 60 characters inclusive.

Diagram



Schema Component Representation

```
<xsd:simpleType name="Token60">
   <xsd:restriction base=" xsd:token ">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="60"/>
   </xsd:restriction>
</xsd:simpleType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: CancelTradeMatch
  - Complex Type: ModifyTradeMatch
  - Complex Type: RequestTradeMatch
  - Complex Type: TradeAlreadyMatched
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target

namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: CancelTradeMatch

[Table of contents]

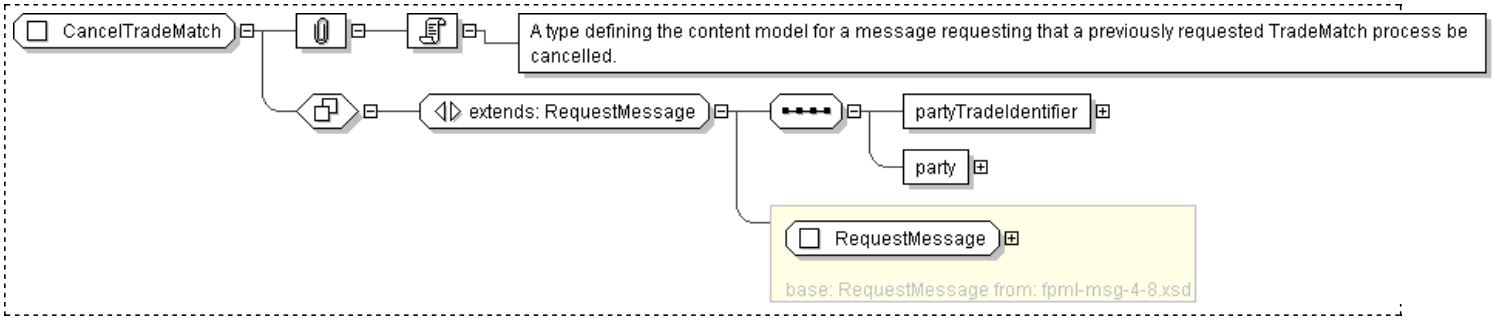
Super-types:	RequestMessage < CancelTradeMatch (by extension)
Sub-types:	None

Name	CancelTradeMatch
Abstract	no
Documentation	A type defining the content model for a message requesting that a previously requested TradeMatch process be cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
 actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <partyTradeIdentifier> PartyTradeIdentifier </partyTradeIdentifier> [1]
   'The trade reference identifier(s) allocated to the trade by the parties involved.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term
   of the trade, but may include other parties involved in, or incidental to, the trade, such as parties
   acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are
   defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CancelTradeMatch">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="partyTradeIdentifier" type=" PartyTradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ModifyTradeMatch

[Table of contents]

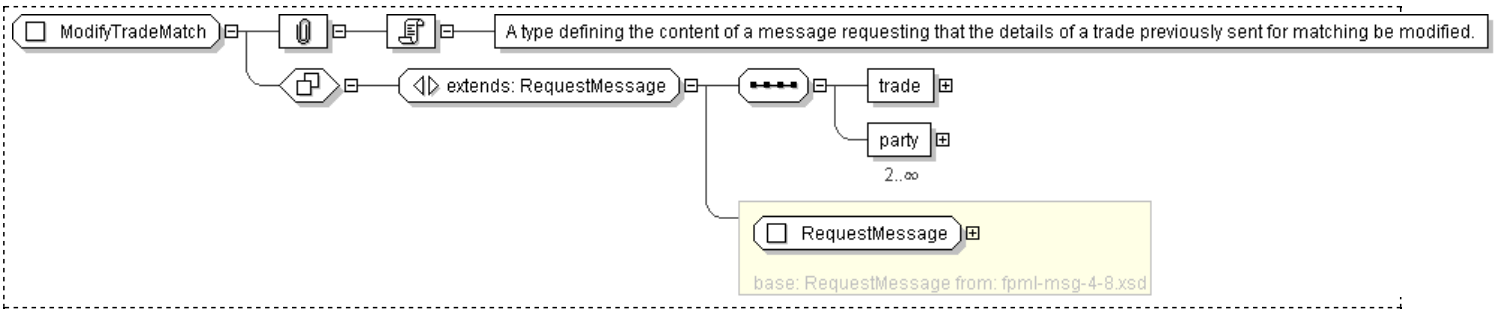
Super-types:	RequestMessage < ModifyTradeMatch (by extension)
Sub-types:	None

Name	ModifyTradeMatch
Abstract	no
Documentation	A type defining the content of a message requesting that the details of a trade previously sent for matching be modified.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term of
   the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in
   the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in
   multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ModifyTradeMatch">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: RequestTradeMatch

[Table of contents]

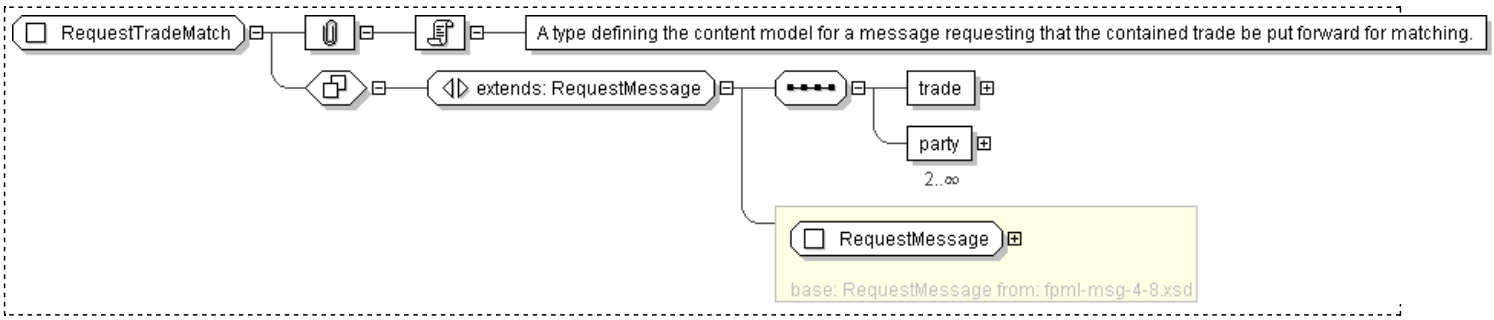
Super-types:	RequestMessage < RequestTradeMatch (by extension)
Sub-types:	None

Name	RequestTradeMatch
Abstract	no
Documentation	A type defining the content model for a message requesting that the contained trade be put forward for matching.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
actual build number stays the same.'

 ">
   <header> RequestMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'The root element in an FpML trade document.'

   <party> Party </party> [2..*]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term
   of the trade, but may include other parties involved in, or incidental to, the trade, such as parties
   acting in the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are
   defined in multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="RequestTradeMatch">
  <xsd:complexContent>
    <xsd:extension base=" RequestMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAlreadyMatched

[Table of contents]

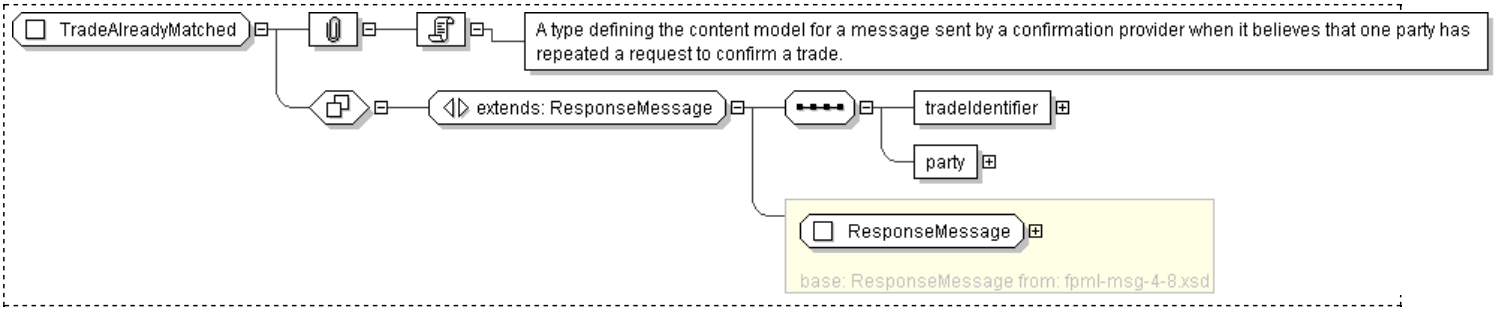
Super-types:	ResponseMessage < TradeAlreadyMatched (by extension)
Sub-types:	None

Name	TradeAlreadyMatched
Abstract	no
Documentation	A type defining the content model for a message sent by a confirmation provider when it believes that one party has repeated a request to confirm a trade.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number
 of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
 the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
 examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
 made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
 number stays the same.'

 ">
   <header> ResponseMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1]
   'An instance of a unique trade identifier.'

   <party> Party </party> [1]
   'A legal entity or a subdivision of a legal entity.','Parties can perform multiple roles in a trade
   lifecycle. For example, the principal parties obligated to make payments from time to time during the term of
   the trade, but may include other parties involved in, or incidental to, the trade, such as parties acting in
   the role of novation transferor/transferee, broker, calculation agent, etc. In FpML roles are defined in
   multiple places within a document.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAlreadyMatched">
  <xsd:complexContent>
    <xsd:extension base=" ResponseMessage ">
      <xsd:sequence>
        <xsd:element name="tradeIdentifier" type=" TradeIdentifier "/>
        <xsd:element name="party" type=" Party "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Definitions
  - Complex Type: TradeAmended
  - Complex Type: TradeCancelled
  - Complex Type: TradeCreated
  - Complex Type: TradeExecution
  - Complex Type: TradeExecutionCancelled
  - Complex Type: TradeExecutionModified
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-msg-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
xml	http://www.w3.org/XML/1998/namespace
ecore	http://www.eclipse.org/emf/2002/Ecore
fpml-annotation	http://www.fpml.org/annotation
fpml	http://www.fpml.org/2010/FpML-4-8
dsig	http://www.w3.org/2000/09/xmldsig#
xsd	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-msg-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in bold.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
      </extension>
   </complexContent>
```

```
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeAmended

[Table of contents]

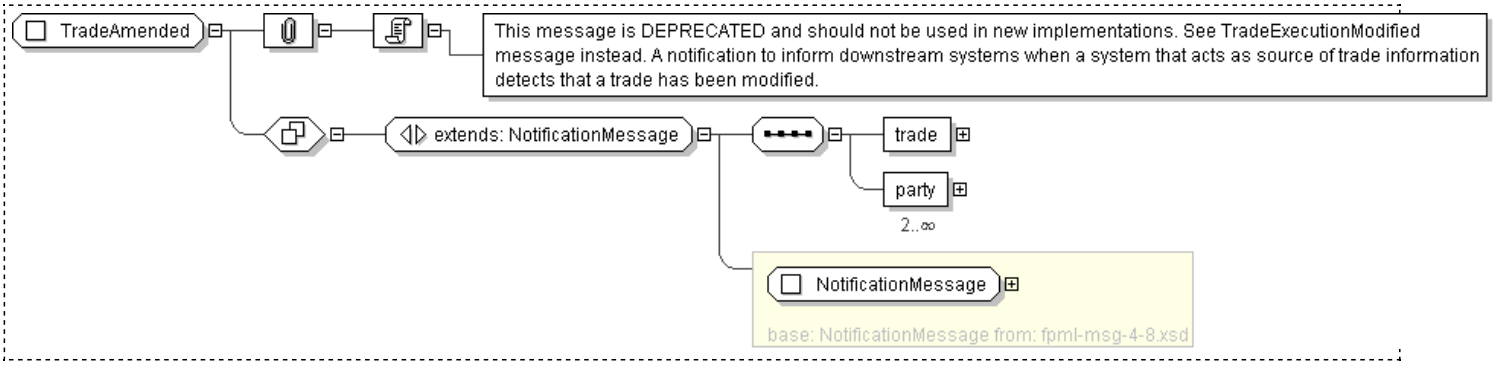
Super-types:	NotificationMessage < TradeAmended (by extension)
Sub-types:	None

Name	TradeAmended
Abstract	no
Documentation	This message is DEPRECATED and should not be used in new implementations. See TradeExecutionModified message instead. A notification to inform downstream systems when a system that acts as source of trade information detects that a trade has been modified.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeAmended" deprecated="true" deprecatedReason="Misuse of this message was causing
interoperability problems. See TradeExecutionModified message instead.">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCancelled

[Table of contents]

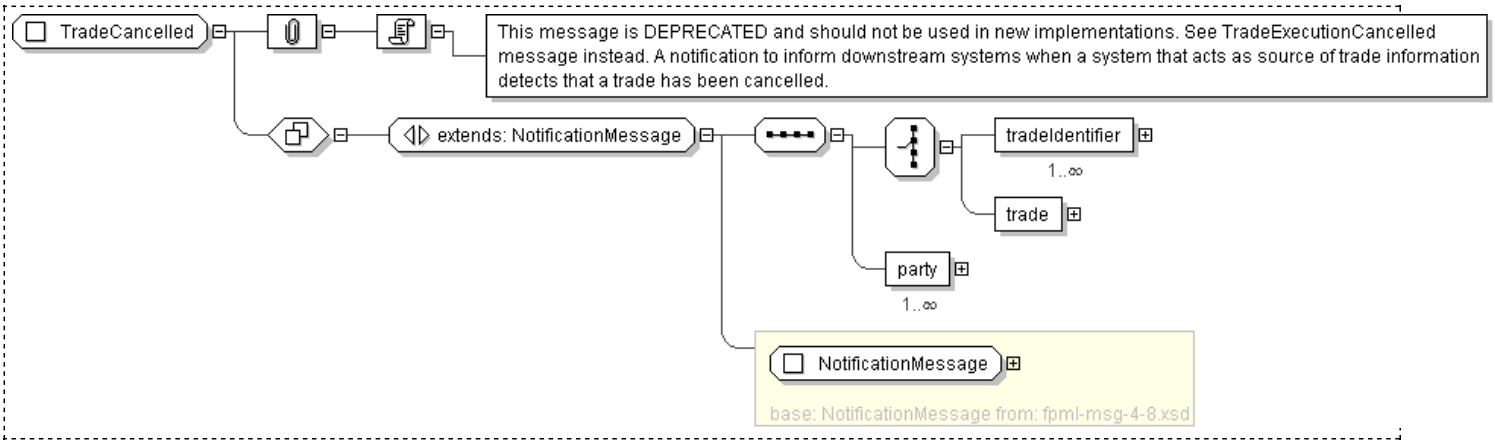
Super-types:	NotificationMessage < TradeCancelled (by extension)
Sub-types:	None

Name	TradeCancelled
Abstract	no
Documentation	This message is DEPRECATED and should not be used in new implementations. See TradeExecutionCancelled message instead. A notification to inform downstream systems when a system that acts as source of trade information detects that a trade has been cancelled.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <tradeIdentifier> TradeIdentifier </tradeIdentifier> [1..*]
      <trade> Trade </trade> [1]
   End Choice
   <party> Party </party> [1..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCancelled" deprecated="true" deprecatedReason="Misuse of this message was causing
interoperability problems. See TradeExecutionCancelled message instead.">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:choice>
               <xsd:element name="tradeIdentifier" type=" TradeIdentifier " maxOccurs="unbounded"/>
               <xsd:element name="trade" type=" Trade "/>
            </xsd:choice>
            <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeCreated

[Table of contents]

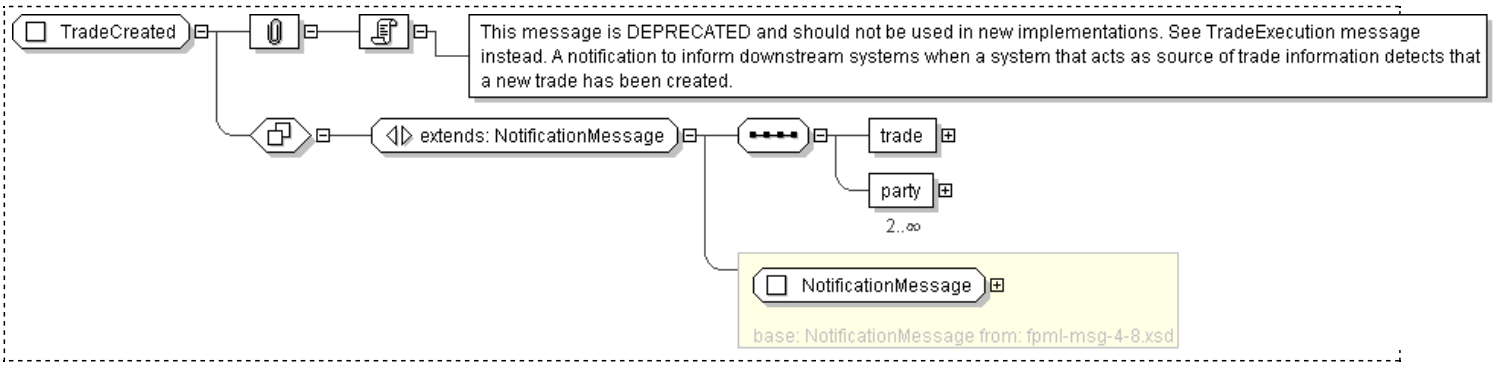
Super-types:	NotificationMessage < TradeCreated (by extension)
Sub-types:	None

Name	TradeCreated
Abstract	no
Documentation	This message is DEPRECATED and should not be used in new implementations. See TradeExecution message instead. A notification to inform downstream systems when a system that acts as source of trade information detects that a new trade has been created.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
'The specific build number of this schema version. This attribute is not included in an instance document.
Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates
the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or
examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been
made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build
number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   <party> Party </party> [2..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeCreated" deprecated="true" deprecatedReason="Misuse of this message was causing
interoperability problems. See TradeExecution message instead for block-level trade execution.">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:element name="trade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeExecution

[Table of contents]

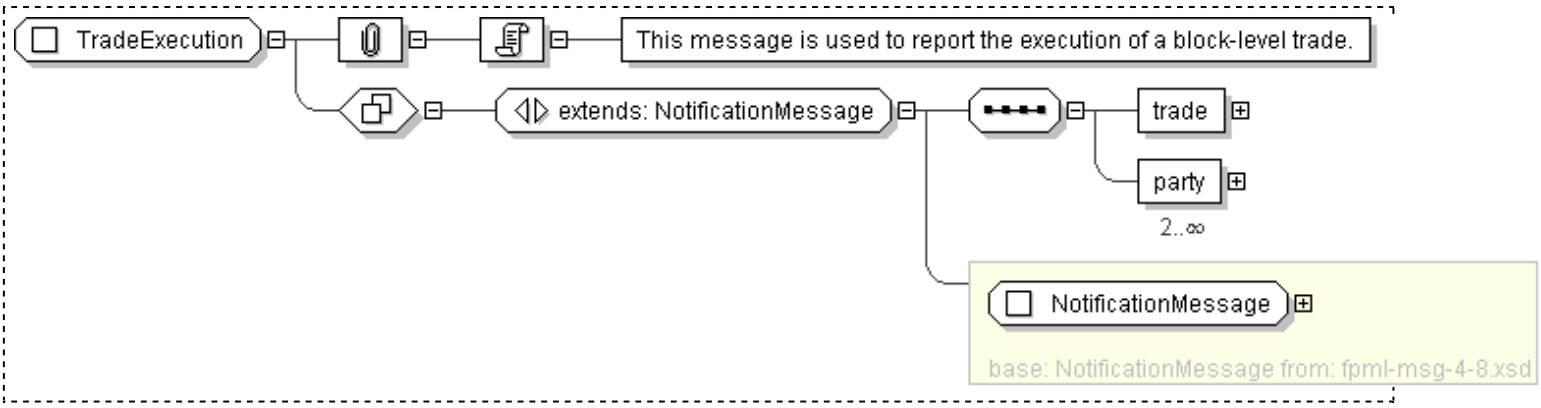
Super-types:	NotificationMessage < TradeExecution (by extension)
Sub-types:	None

Name	TradeExecution
Abstract	no
Documentation	This message is used to report the execution of a block-level trade.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   <trade> Trade </trade> [1]
   'Block-level trade being executed.'

   <party> Party </party> [2..*]
   'Parties referenced by the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeExecution">
   <xsd:complexContent>
      <xsd:extension base=" NotificationMessage ">
         <xsd:sequence>
            <xsd:element name="trade" type=" Trade "/>
            <xsd:element name="party" type=" Party " minOccurs="2"
            maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeExecutionCancelled

[Table of contents]

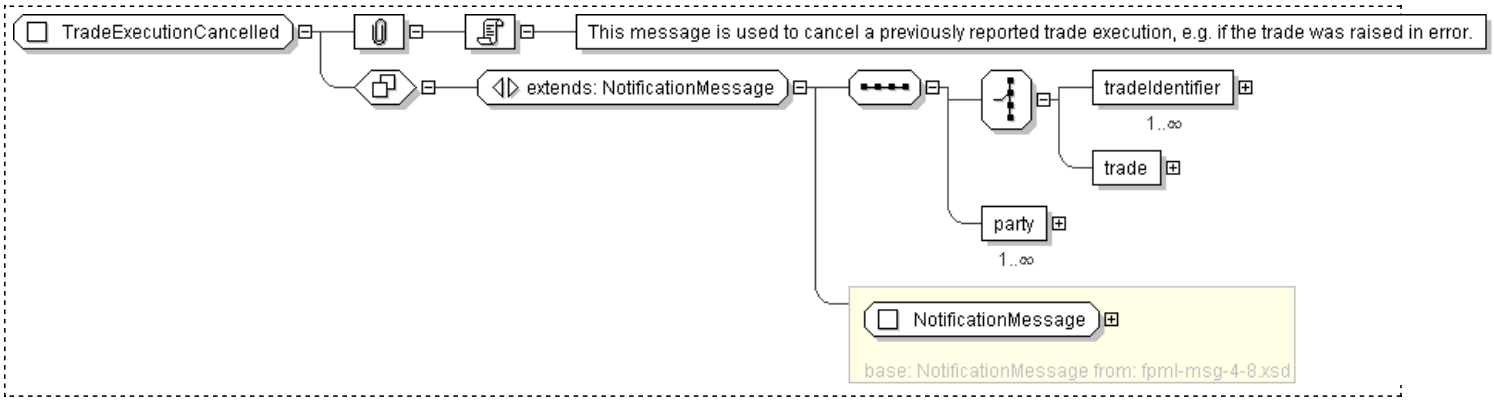
Super-types:	NotificationMessage < TradeExecutionCancelled (by extension)
Sub-types:	None

Name	TradeExecutionCancelled
Abstract	no
Documentation	This message is used to cancel a previously reported trade execution, e.g. if the trade was raised in error.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build
 number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document.
 Instead, it is supplied by the XML parser when the document is validated against the FpML schema and
 indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation
 rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes
 have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the
 actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [1]
      <tradeIdentifier> PartyTradeIdentifier </tradeIdentifier> [1..*]
      <trade> Trade </trade> [1]
   End Choice
   <party> Party </party> [1..*]
   'Parties referenced by the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeExecutionCancelled">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="tradeIdentifier" type=" PartyTradeIdentifier " maxOccurs="unbounded"/>
          <xsd:element name="trade" type=" Trade "/>
        </xsd:choice>
        <xsd:element name="party" type=" Party " maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: TradeExecutionModified

[Table of contents]

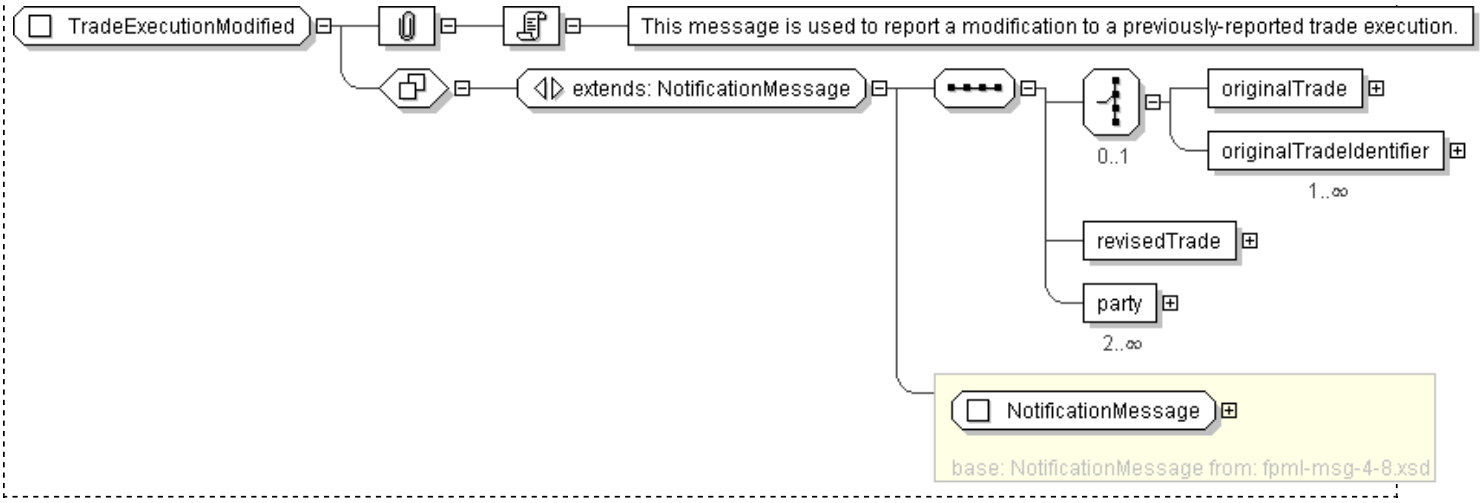
Super-types:	NotificationMessage < TradeExecutionModified (by extension)
Sub-types:	None

Name	TradeExecutionModified
Abstract	no
Documentation	This message is used to report a modification to a previously-reported trade execution.

XML Instance Representation

```
<...
 version=" xsd:token (value comes from list: {'4-0'|'4-1'|'4-2'|'4-3'|'4-4'|'4-5'|'4-6'|'4-7'|'4-8'}) [1]
 'Indicate which version of the FpML Schema an FpML message adheres to.'

 "
 expectedBuild=" xsd:positiveInteger [0..1]
 'This optional attribute can be supplied by a message creator in an FpML instance to specify which build number of the schema was used to define the message when it was generated.'

 "
 actualBuild="3 [0..1]
 'The specific build number of this schema version. This attribute is not included in an instance document. Instead, it is supplied by the XML parser when the document is validated against the FpML schema and indicates the build number of the schema file. Every time FpML publishes a change to the schema, validation rules, or examples within a version (e.g., version 4.2) the actual build number is incremented. If no changes have been made between releases within a version (i.e. from Trial Recommendation to Recommendation) the actual build number stays the same.'

 ">
   <header> NotificationMessageHeader </header> [1]
   <validation> Validation </validation> [0..*]
   Start Choice [0..1]
      <originalTrade> Trade </originalTrade> [1]
      'Complete description of the trade prior to the modification.'

      <originalTradeIdentifier> PartyTradeIdentifier </originalTradeIdentifier> [1..*]
      'Identifiers of the trade prior to the modification.'

   End Choice
   <revisedTrade> Trade </revisedTrade> [1]
   'Complete description of the trade after the modification.'

   <party> Party </party> [2..*]
   'Parties referenced by the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="TradeExecutionModified">
  <xsd:complexContent>
    <xsd:extension base=" NotificationMessage ">
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="originalTrade" type=" Trade "/>
          <xsd:element name="originalTradeIdentifier" type=" PartyTradeIdentifier "
          maxOccurs="unbounded"/>
        </xsd:choice>
        <xsd:element name="revisedTrade" type=" Trade "/>
        <xsd:element name="party" type=" Party " minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: valuationSet
- Global Definitions
  - Complex Type: AssetValuation
  - Complex Type: DerivedValuationScenario
  - Complex Type: Position
  - Complex Type: PositionConstituent
  - Complex Type: Quotation
  - Complex Type: ReportingRoles
  - Complex Type: ScheduledDate
  - Complex Type: ScheduledDateType
  - Complex Type: ScheduledDates
  - Complex Type: Sensitivity
  - Complex Type: SensitivitySet
  - Complex Type: ValuationSet
  - Complex Type: ValuationSetDetail
  - Complex Type: Valuations
  - Model Group: AdjustedAndOrUnadjustedDate.model
  - Model Group: AssetValuationOrReference.model
  - Model Group: AssociatedValue.model
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 6991 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-riskdef-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 6991 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-riskdef-4-8.xsd"/>
   ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
```

```
        <sequence>
           <element name="state" type=" AusStates "/>
           <element name="postcode">
              <simpleType>
                 <restriction base=" string ">
                    <pattern value="[1-9][0-9]{3}"/>
                 </restriction>
              </simpleType>
           </element>
        </sequence>
        <attribute name="country" type=" string " fixed="Australia"/>
     </extension>
   </complexContent>
</complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

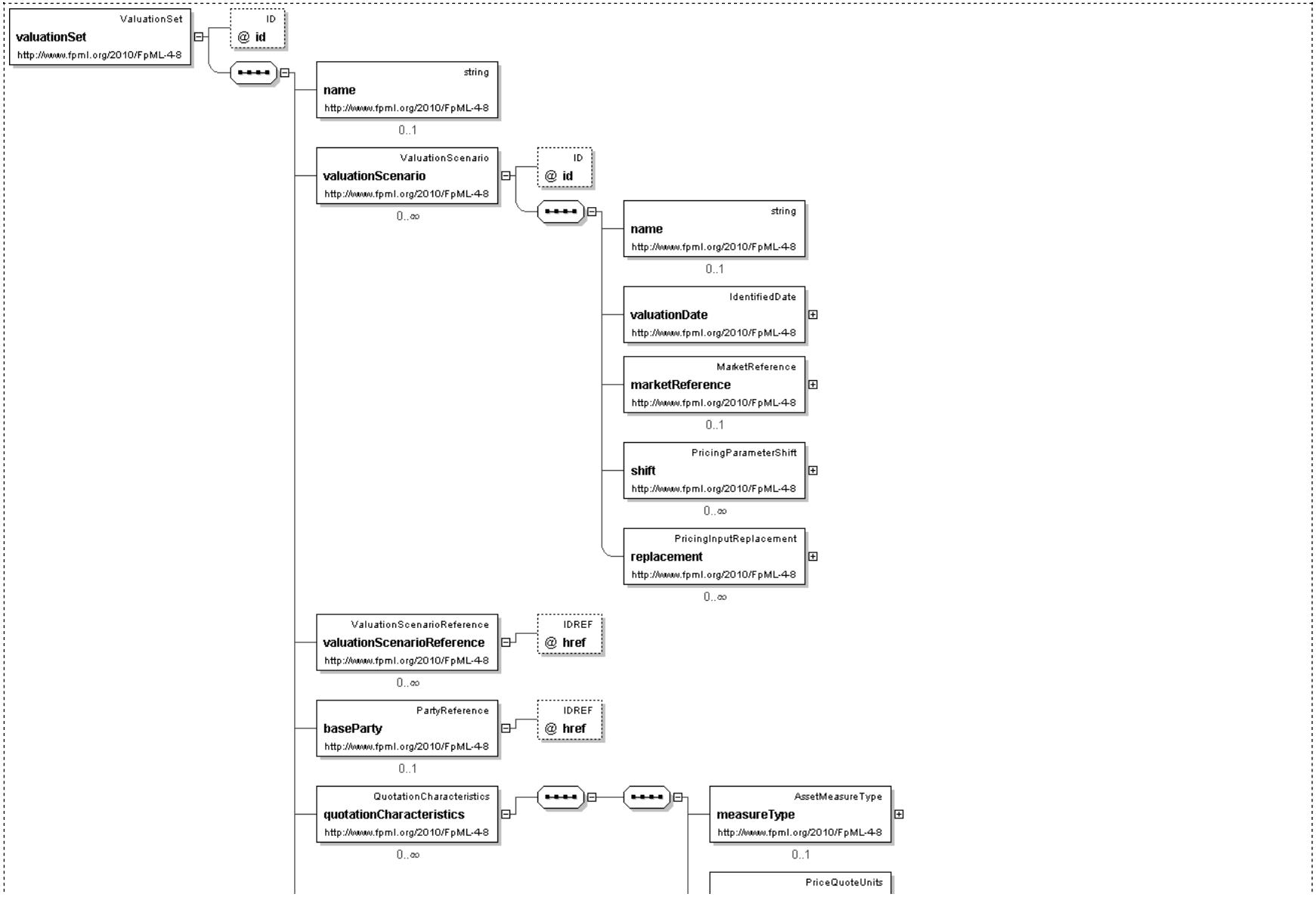
XML Schema Documentation

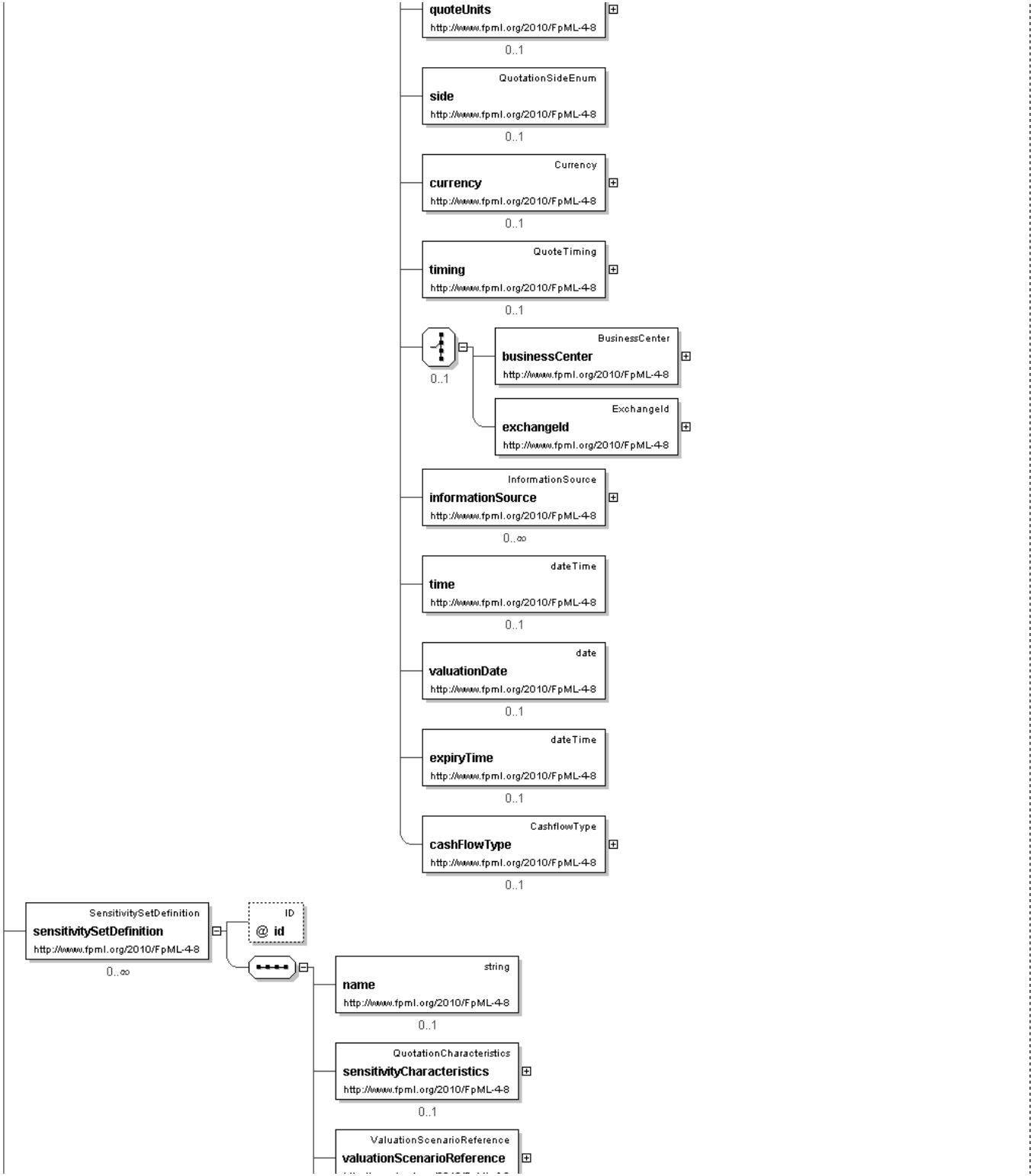
Element: valuationSet

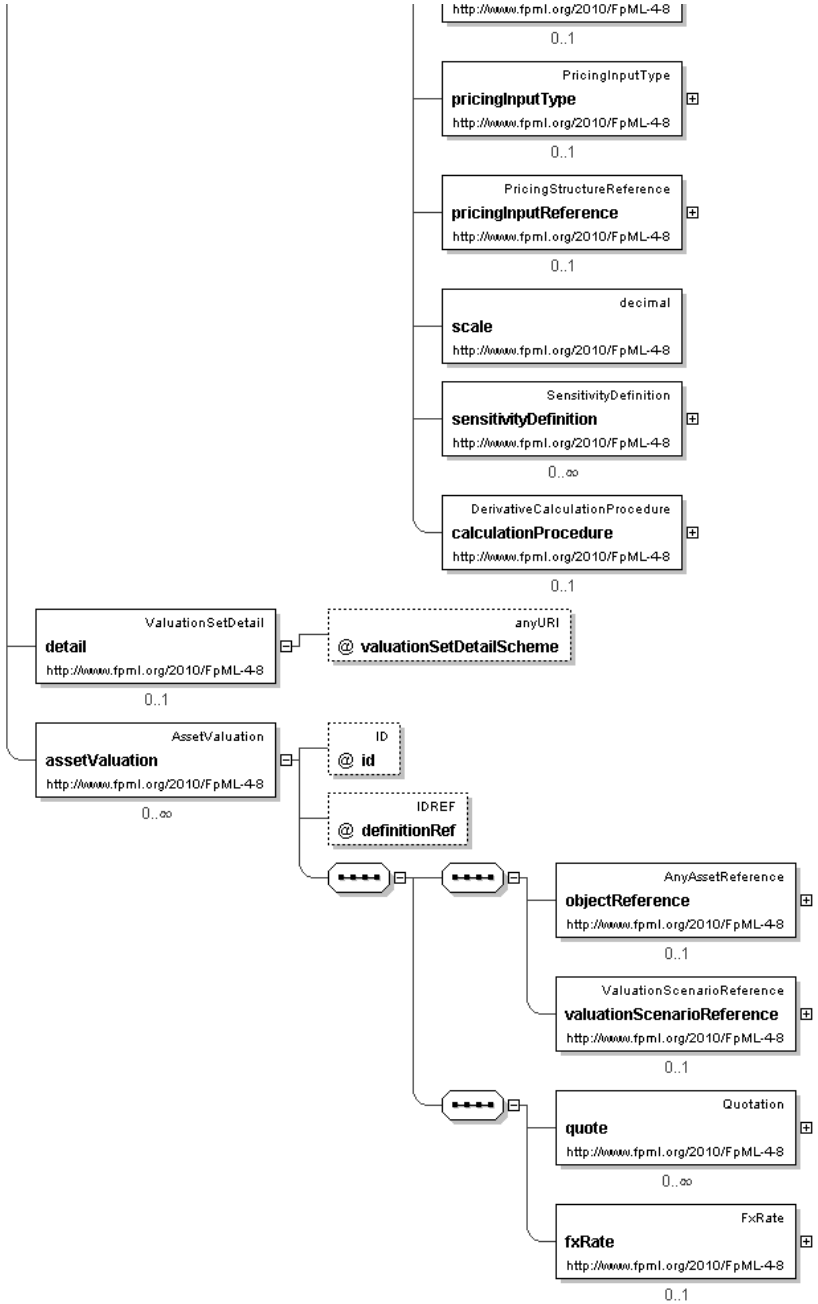
[Table of contents]

Name	valuationSet
Type	ValuationSet
Nillable	no
Abstract	no

Logical Diagram




quoteUnits
http://www.fpml.org/2010/FpML-4-8
0..1
QuotationSideEnum
side
http://www.fpml.org/2010/FpML-4-8
0..1
Currency
currency
http://www.fpml.org/2010/FpML-4-8
0..1
QuoteTiming
timing
http://www.fpml.org/2010/FpML-4-8
0..1
0..1
BusinessCenter
businessCenter
http://www.fpml.org/2010/FpML-4-8
ExchangeId
exchangeId
http://www.fpml.org/2010/FpML-4-8
InformationSource
informationSource
http://www.fpml.org/2010/FpML-4-8
0..∞
dateTime
time
http://www.fpml.org/2010/FpML-4-8
0..1
date
valuationDate
http://www.fpml.org/2010/FpML-4-8
0..1
dateTime
expiryTime
http://www.fpml.org/2010/FpML-4-8
0..1
CashflowType
cashFlowType
http://www.fpml.org/2010/FpML-4-8
0..1
SensitivitySetDefinition
sensitivitySetDefinition
http://www.fpml.org/2010/FpML-4-8
0..∞
ID
@ id
string
name
http://www.fpml.org/2010/FpML-4-8
0..1
QuotationCharacteristics
sensitivityCharacteristics
http://www.fpml.org/2010/FpML-4-8
0..1
ValuationScenarioReference
valuationScenarioReference



XML Instance Representation

```
<valuationSet
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The name of the valuation set, used to understand what it means. E.g., \"EOD Values and Risks for Party A\".'

   <valuationScenario> ValuationScenario </valuationScenario> [0..*]
   'Valuation scenerios used (requested/reported) in this valuation set. E.g., the EOD valuation scenario for a particular value date. Used for the first
   occurrence of a valuation scenario in a document.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..*]
```

'References to valuation scenarios used (requested/reported) in this valuation set. E..g, a reference to the EOD valuation scenario for a particular value date. Used for subsequence occurrences of a valuation set in an FpML document.'

<baseParty> PartyReference </baseParty> [0..1]
'Reference to the party from whose point of view the assets are valued.'

<quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..*]
'Charactistics (measure types, units, sides, etc.) of the quotes used (requested/reported) in the valuation set.'

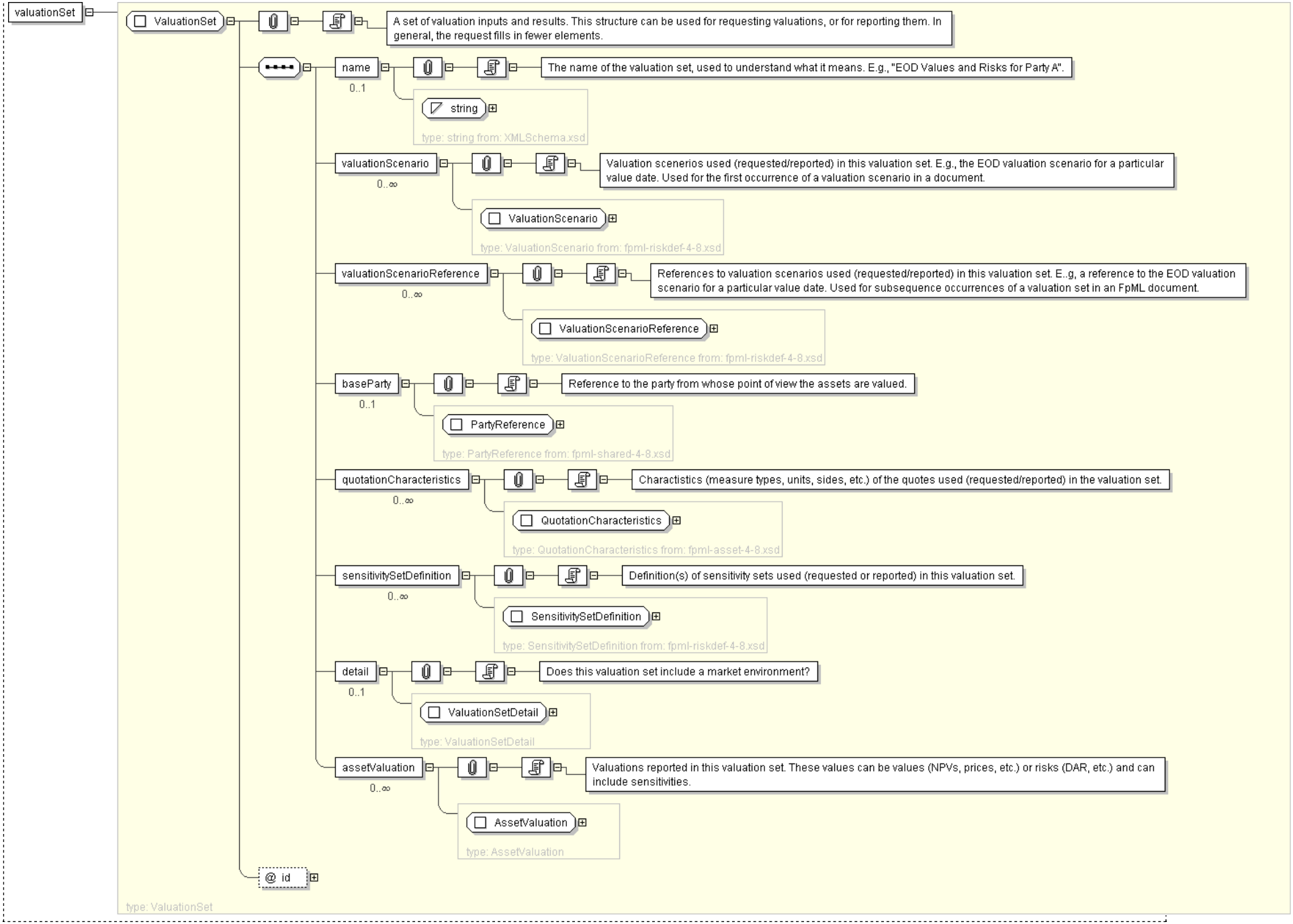
<sensitivitySetDefinition> SensitivitySetDefinition </sensitivitySetDefinition> [0..*]
'Definition(s) of sensitivity sets used (requested or reported) in this valuation set.'

<detail> ValuationSetDetail </detail> [0..1]
'Does this valuation set include a market environment?'

<assetValuation> AssetValuation </assetValuation> [0..*]
'Valuations reported in this valuation set. These values can be values (NPVs, prices, etc.) or risks (DAR, etc.) and can include sensitivities.'

</valuationSet>

Diagram



Schema Component Representation

```
<xsd:element name="valuationSet" type=" ValuationSet "/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AdjustedAndOrUnadjustedDate.model

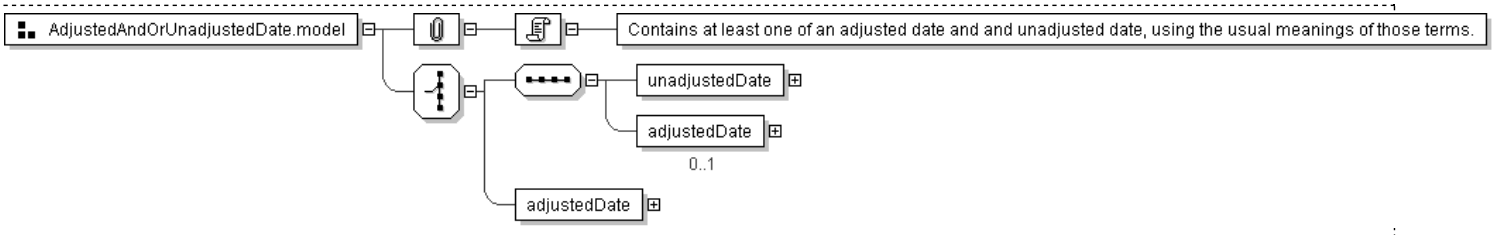
[Table of contents]

Name	AdjustedAndOrUnadjustedDate.model
Used by (from the same schema document)	Complex Type ScheduledDate
Documentation	Contains at least one of an adjusted date and and unadjusted date, using the usual meanings of those terms.

XML Instance Representation

```
Start Choice [1]
   <unadjustedDate> xsd:date </unadjustedDate> [1]
   <adjustedDate> xsd:date </adjustedDate> [0..1]
   <adjustedDate> xsd:date </adjustedDate> [1]
End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="AdjustedAndOrUnadjustedDate.model">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="unadjustedDate" type=" xsd:date "/>
      <xsd:element name="adjustedDate" type=" xsd:date " minOccurs="0"/>
    </xsd:sequence>
    <xsd:element name="adjustedDate" type=" xsd:date "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AssetValuationOrReference.model

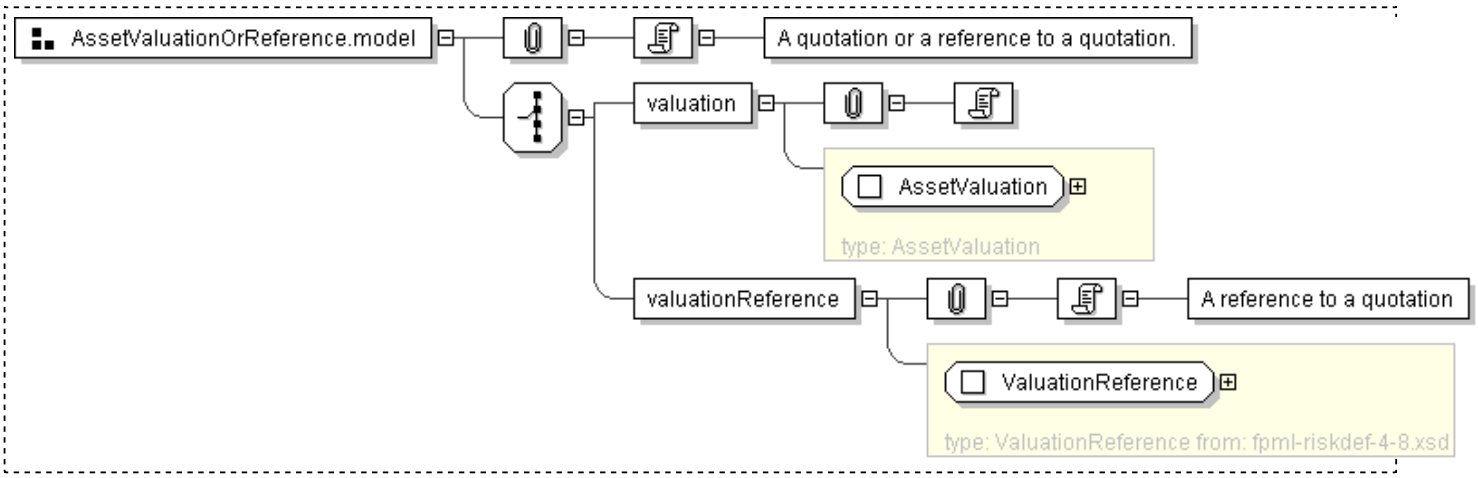
[Table of contents]

Name	AssetValuationOrReference.model
Used by (from the same schema document)	Complex Type Valuations
Documentation	A quotation or a reference to a quotation.

XML Instance Representation

```
Start Choice [1]
   <valuation> AssetValuation </valuation> [1]
   ''

   <valuationReference> ValuationReference </valuationReference> [1]
   'A reference to a quotation'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="AssetValuationOrReference.model">
  <xsd:choice>
    <xsd:element name="valuation" type=" AssetValuation "/>
    <xsd:element name="valuationReference" type=" ValuationReference "/>
  </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Model Group: AssociatedValue.model

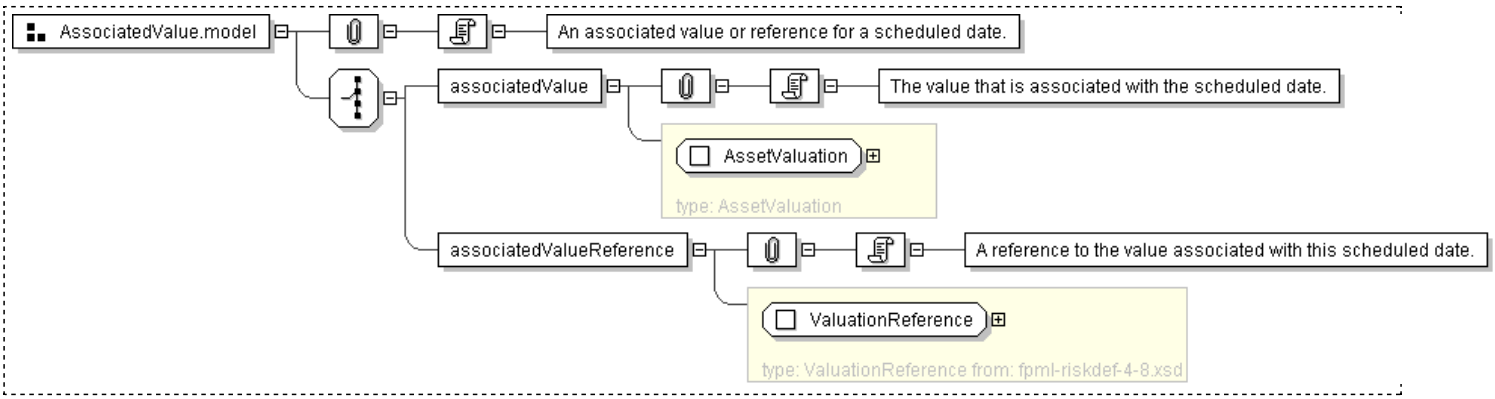
[Table of contents]

Name	AssociatedValue.model
Used by (from the same schema document)	Complex Type ScheduledDate
Documentation	An associated value or reference for a scheduled date.

XML Instance Representation

```
Start Choice [1]
   <associatedValue> AssetValuation </associatedValue> [1]
   'The value that is associated with the scheduled date.'

   <associatedValueReference> ValuationReference </associatedValueReference> [1]
   'A reference to the value associated with this scheduled date.'

End Choice
```

Diagram



Schema Component Representation

```
<xsd:group name="AssociatedValue.model">
   <xsd:choice>
      <xsd:element name="associatedValue" type=" AssetValuation "/>
      <xsd:element name="associatedValueReference" type=" ValuationReference "/>
   </xsd:choice>
</xsd:group>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: AssetValuation

[Table of contents]

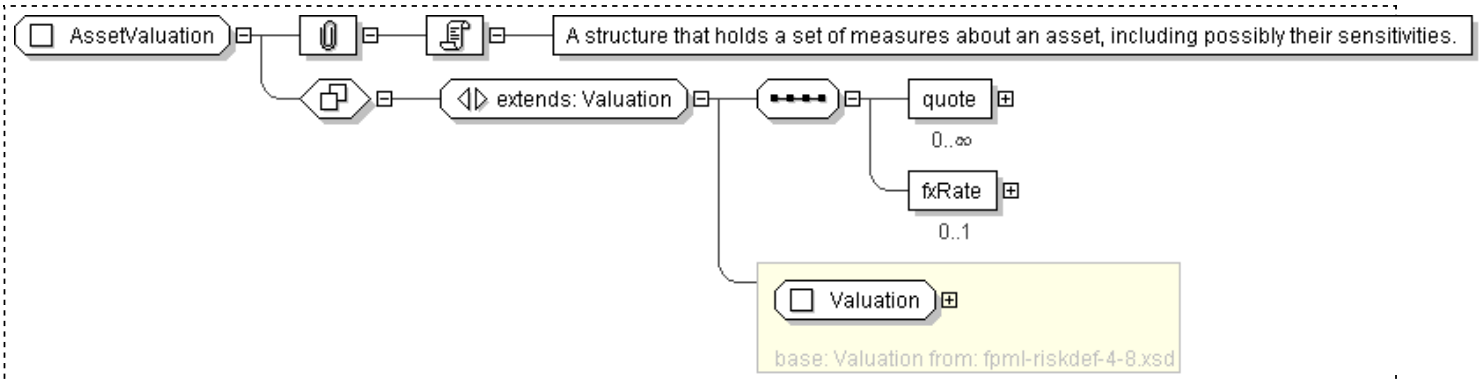
Super-types:	Valuation < AssetValuation (by extension)
Sub-types:	None

Name	AssetValuation
Used by (from the same schema document)	Complex Type Position , Complex Type ValuationSet , Model Group AssetValuationOrReference.model , Model Group AssociatedValue.model
Abstract	no
Documentation	A structure that holds a set of measures about an asset, including possibly their sensitivities.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]"
 definitionRef=" xsd:IDREF [0..1]
'An optional reference to the scenario that this valuation applies to.'

 ">
   <objectReference> AnyAssetReference </objectReference> [0..1]
   'A reference to the asset or pricing structure that this values.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..1]
   'A reference to the valuation scenario used to calculate this valuation. If the Valuation
   occurs within a ValuationSet, this value is optional and is defaulted from the ValuationSet.
   If this value occurs in both places, the lower level value (i.e. the one here) overrides
   that in the higher (i.e. ValuationSet).'

   <quote> Quotation </quote> [0..*]
   'One or more numerical measures relating to the asset, possibly together with sensitivities
   of that measure to pricing inputs.'

   <fxRate> FxRate </fxRate> [0..1]
   'Indicates the rate of a currency conversion that may have been used to compute
   valuations.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AssetValuation">
  <xsd:complexContent>
    <xsd:extension base=" Valuation ">
      <xsd:sequence>
        <xsd:element name="quote" type=" Quotation " minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="fxRate" type=" FxRate " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
```

```
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: DerivedValuationScenario

[Table of contents]

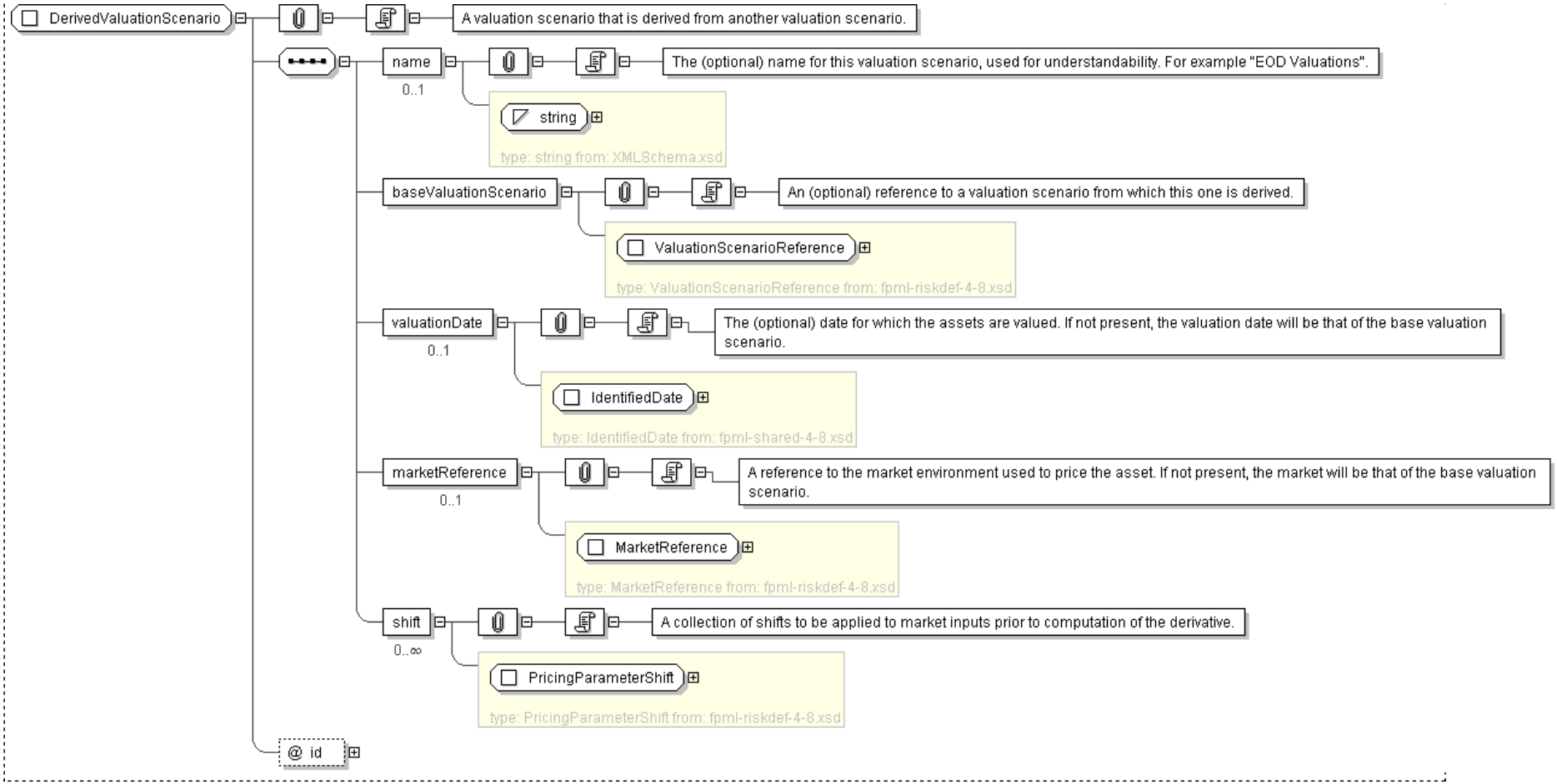
Super-types:	None
Sub-types:	None

Name	DerivedValuationScenario
Abstract	no
Documentation	A valuation scenario that is derived from another valuation scenario.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The (optional) name for this valuation scenario, used for understandability. For example \"EOD Valuations\".'

   <baseValuationScenario> ValuationScenarioReference </baseValuationScenario> [1]
   'An (optional) reference to a valuation scenario from which this one is derived.'

   <valuationDate> IdentifiedDate </valuationDate> [0..1]
   'The (optional) date for which the assets are valued. If not present, the valuation date will be that of the base valuation scenario.'

   <marketReference> MarketReference </marketReference> [0..1]
   'A reference to the market environment used to price the asset. If not present, the market will be that of the base valuation scenario.'

   <shift> PricingParameterShift </shift> [0..*]
   'A collection of shifts to be applied to market inputs prior to computation of the derivative.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DerivedValuationScenario">
  <xsd:sequence>
    <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
    <xsd:element name="baseValuationScenario" type=" ValuationScenarioReference "/>
    <xsd:element name="valuationDate" type=" IdentifiedDate " minOccurs="0"/>
    <xsd:element name="marketReference" type=" MarketReference " minOccurs="0"/>
    <xsd:element name="shift" type=" PricingParameterShift " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Position

[Table of contents]

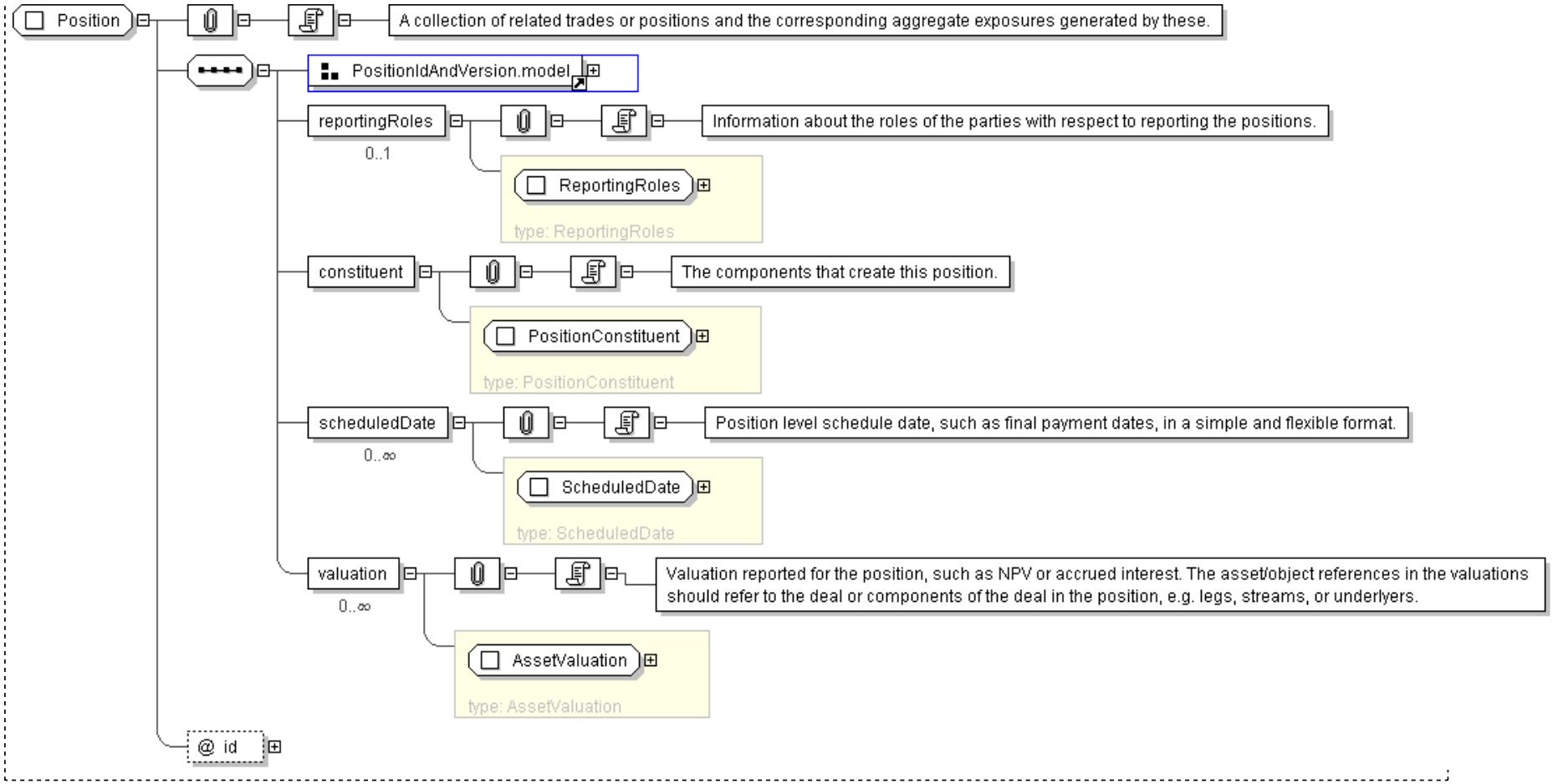
Super-types:	None
Sub-types:	None

Name	Position
Abstract	no
Documentation	A collection of related trades or positions and the corresponding aggregate exposures generated by these.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <positionId> PositionId </positionId> [1]
    'A version-independent identifier for the position, possibly based on trade identifier.'

   <version> xsd:positiveInteger </version> [0..1]
    'A version identifier. Version identifiers must be ascending, i.e. higher numbers imply newer versions. There is no
    requirement that version identifiers for a position be sequential or small, so for example timestamp-based version
    identifiers could be used.'

   <reportingRoles> ReportingRoles </reportingRoles> [0..1]
    'Information about the roles of the parties with respect to reporting the positions.'

   <constituent> PositionConstituent </constituent> [1]
    'The components that create this position.'

   <scheduledDate> ScheduledDate </scheduledDate> [0..*]
    'Position level schedule date, such as final payment dates, in a simple and flexible format.'

   <valuation> AssetValuation </valuation> [0..*]
    'Valuation reported for the position, such as NPV or accrued interest. The asset/object references in the valuations
    should refer to the deal or components of the deal in the position, e.g. legs, streams, or underlyers.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Position">
  <xsd:sequence>
    <xsd:group ref=" PositionIdAndVersion.model "/>
    <xsd:element name="reportingRoles" type=" ReportingRoles " minOccurs="0"/>
    <xsd:element name="constituent" type=" PositionConstituent "/>
    <xsd:element name="scheduledDate" type=" ScheduledDate " minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="valuation" type=" AssetValuation " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: PositionConstituent

[Table of contents]

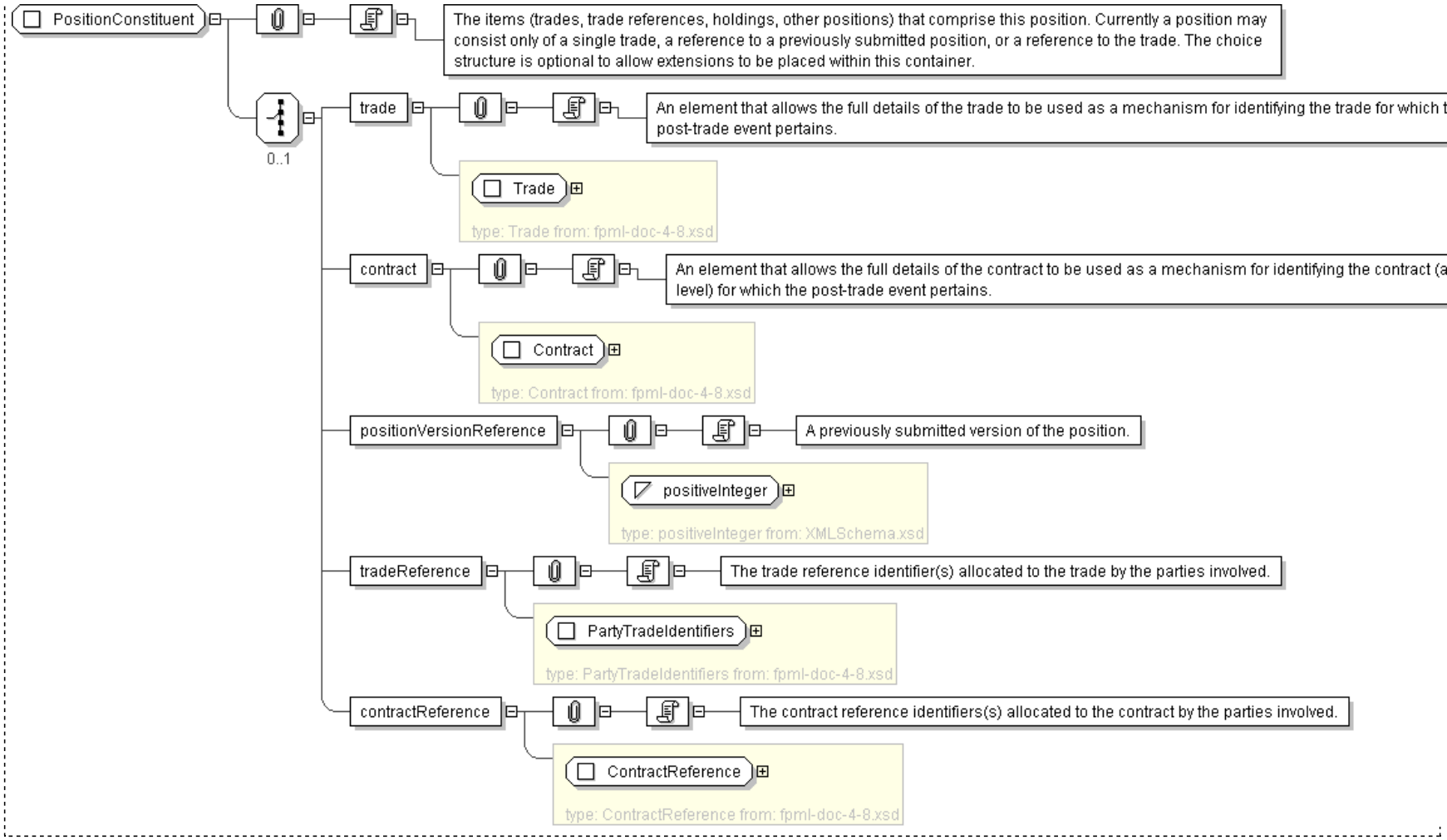
Super-types:	None
Sub-types:	None

Name	PositionConstituent
Used by (from the same schema document)	Complex Type Position
Abstract	no
Documentation	The items (trades, trade references, holdings, other positions) that comprise this position. Currently a position may consist only of a single trade, a reference to a previously submitted position, or a reference to the trade. The choice structure is optional to allow extensions to be placed within this container.

XML Instance Representation

```
<...>
  Start Choice [0..1]
    <trade> Trade </trade> [1]
     'An element that allows the full details of the trade to be used as a mechanism for identifying the trade for which the post-
     trade event pertains.'

    <contract> Contract </contract> [1]
     'An element that allows the full details of the contract to be used as a mechanism for identifying the contract (at allocation
     level) for which the post-trade event pertains.'

    <positionVersionReference> xsd:positiveInteger </positionVersionReference> [1]
     'A previously submitted version of the position.'

    <tradeReference> PartyTradeIdentifiers </tradeReference> [1]
     'The trade reference identifier(s) allocated to the trade by the parties involved.'

    <contractReference> ContractReference </contractReference> [1]
     'The contract reference identifiers(s) allocated to the contract by the parties involved.'

  End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionConstituent">
  <xsd:choice minOccurs="0">
    <xsd:element name="trade" type=" Trade "/>
    <xsd:element name="contract" type=" Contract "/>
    <xsd:element name="positionVersionReference" type=" xsd:positiveInteger "/>
    <xsd:element name="tradeReference" type=" PartyTradeIdentifiers "/>
    <xsd:element name="contractReference" type=" ContractReference "/>
  </xsd:choice>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Quotation

[Table of contents]

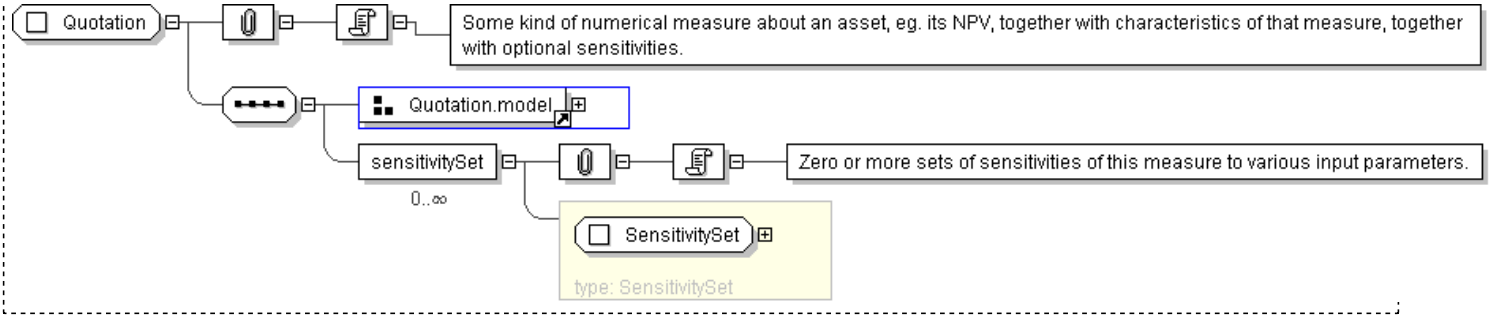
Super-types:	None
Sub-types:	None

Name	Quotation
Used by (from the same schema document)	Complex Type AssetValuation
Abstract	no
Documentation	Some kind of numerical measure about an asset, eg. its NPV, together with characteristics of that measure, together with optional sensitivities.

XML Instance Representation

```
<...>
   <value> xsd:decimal </value> [0..1]
   'The value of the the quotation.'

   <measureType> AssetMeasureType </measureType> [0..1]
   'The type of the value that is measured. This could be an NPV, a cash flow, a clean price, etc.'

   <quoteUnits> PriceQuoteUnits </quoteUnits> [0..1]
   'The optional units that the measure is expressed in. If not supplied, this is assumed to be a
   price/value in currency units.'

   <side> QuotationSideEnum </side> [0..1]
   'The side (bid/mid/ask) of the measure.'

   <currency> Currency </currency> [0..1]
   'The optional currency that the measure is expressed in. If not supplied, this is defaulted from the
   reportingCurrency in the valuationScenarioDefinition.'

   <timing> QuoteTiming </timing> [0..1]
   'When during a day the quote is for. Typically, if this element is supplied, the QuoteLocation needs
   also to be supplied.'

   Start Group: QuoteLocation.model [0..1]
'Where the quote is from.'


      Start Choice [1]
         <businessCenter> BusinessCenter </businessCenter> [1]
         'A city or other business center.'

         <exchangeId> ExchangeId </exchangeId> [1]
         'The exchange (e.g. stock or futures exchange) from which the quote is obtained.'

      End Choice
   End Group: QuoteLocation.model
   <informationSource> InformationSource </informationSource> [0..*]
   'The information source where a published or displayed market rate will be obtained, e.g. Telerate Page
   3750.'

   <time> xsd:dateTime </time> [0..1]
   'When the quote was observed or derived.'

   <valuationDate> xsd:date </valuationDate> [0..1]
   'When the quote was computed.'

   <expiryTime> xsd:dateTime </expiryTime> [0..1]
   'When does the quote cease to be valid.'

   <cashFlowType> CashflowType </cashFlowType> [0..1]
   'For cash flows, the type of the cash flows. Examples include: Coupon payment, Premium Fee, Settlement
   Fee, Brokerage Fee, etc.'

   <sensitivitySet> SensitivitySet </sensitivitySet> [0..*]
   'Zero or more sets of sensitivities of this measure to various input parameters.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Quotation">
  <xsd:sequence>
    <xsd:group ref=" Quotation.model "/>
    <xsd:element name="sensitivitySet" type=" SensitivitySet " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ReportingRoles

[Table of contents]

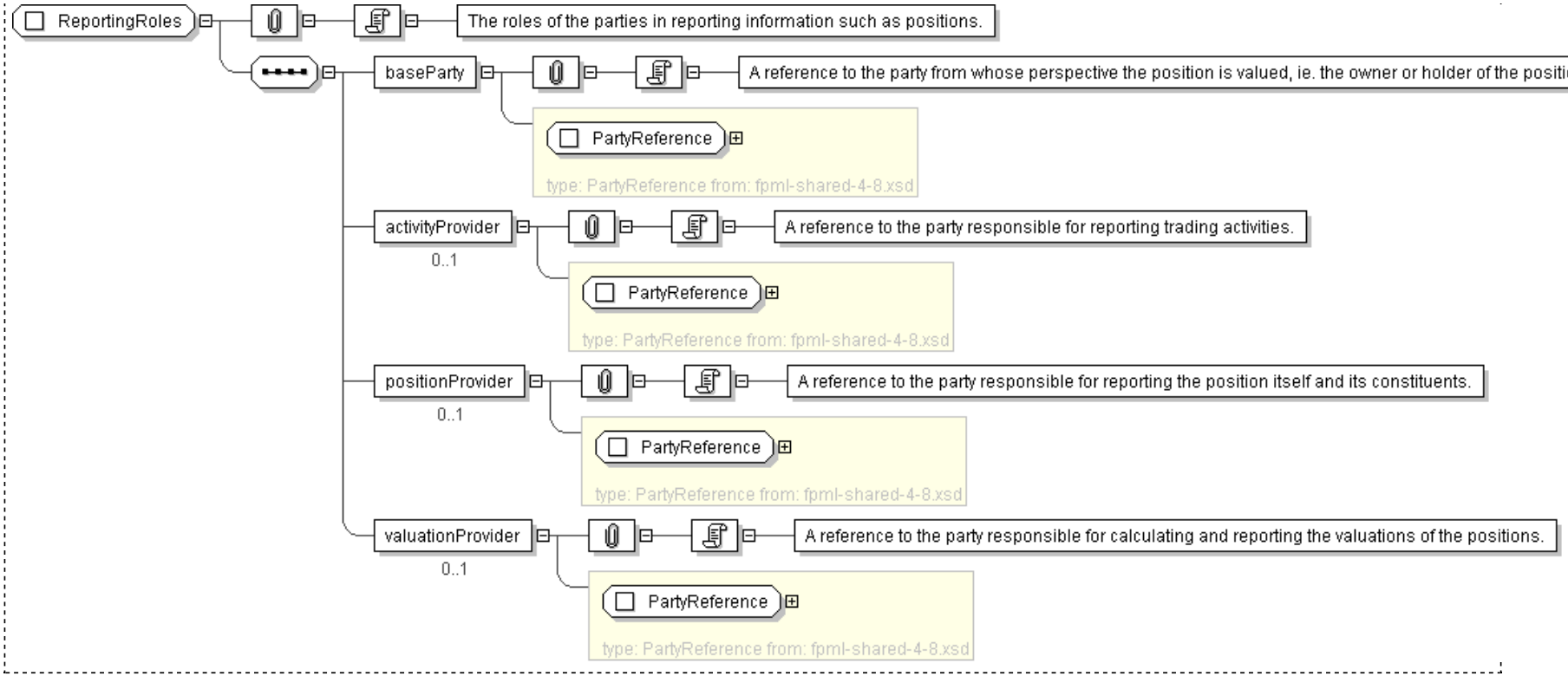
Super-types:	None
Sub-types:	None

Name	ReportingRoles
Used by (from the same schema document)	Complex Type Position
Abstract	no
Documentation	The roles of the parties in reporting information such as positions.

XML Instance Representation

```
<...>
   <baseParty> PartyReference </baseParty> [1]
   'A reference to the party from whose perspective the position is valued, ie. the owner or holder of the position.'

   <activityProvider> PartyReference </activityProvider> [0..1]
   'A reference to the party responsible for reporting trading activities.'

   <positionProvider> PartyReference </positionProvider> [0..1]
   'A reference to the party responsible for reporting the position itself and its constituents.'

   <valuationProvider> PartyReference </valuationProvider> [0..1]
   'A reference to the party responsible for calculating and reporting the valuations of the positions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReportingRoles">
  <xsd:sequence>
    <xsd:element name="baseParty" type=" PartyReference "/>
    <xsd:element name="activityProvider" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="positionProvider" type=" PartyReference " minOccurs="0"/>
    <xsd:element name="valuationProvider" type=" PartyReference " minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ScheduledDate

[Table of contents]

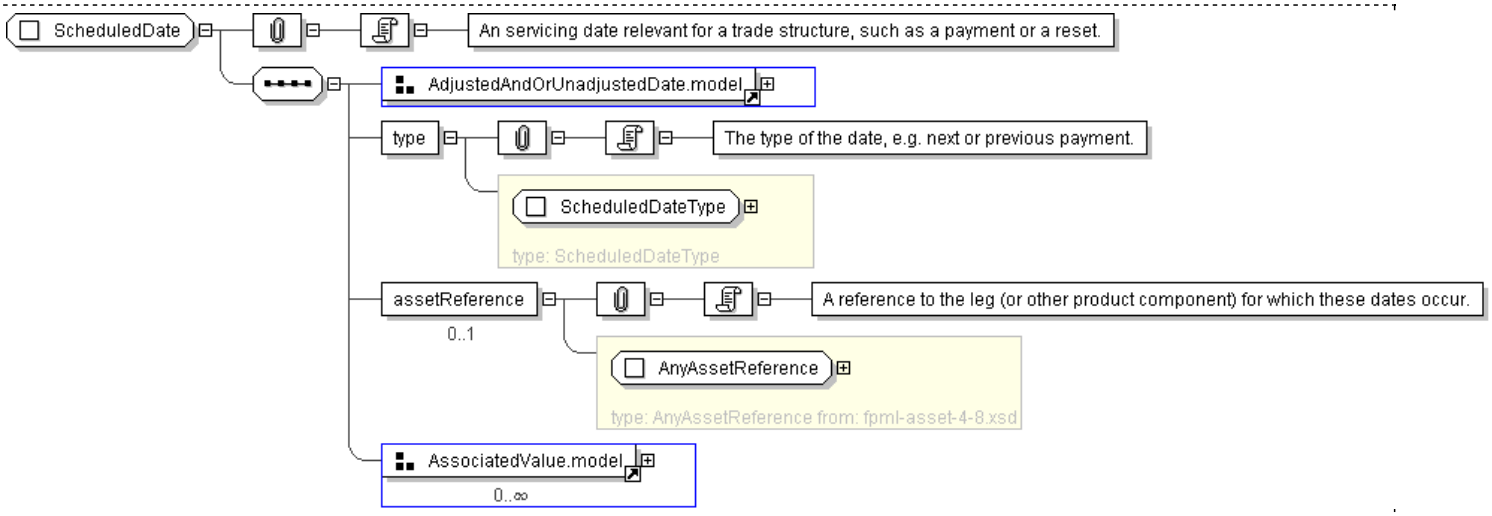
Super-types:	None
Sub-types:	None

Name	ScheduledDate
Used by (from the same schema document)	Complex Type Position , Complex Type ScheduledDates
Abstract	no
Documentation	An servicing date relevant for a trade structure, such as a payment or a reset.

XML Instance Representation

```
<...>
  Start Choice [1]
     <unadjustedDate> xsd:date </unadjustedDate> [1]
     <adjustedDate> xsd:date </adjustedDate> [0..1]
     <adjustedDate> xsd:date </adjustedDate> [1]
  End Choice
  <type> ScheduledDateType </type> [1]
  'The type of the date, e.g. next or previous payment.'

  <assetReference> AnyAssetReference </assetReference> [0..1]
  'A reference to the leg (or other product component) for which these dates occur.'

  Start Group: AssociatedValue.model [0..*]
     Start Choice [1]
        <associatedValue> AssetValuation </associatedValue> [1]
        'The value that is associated with the scheduled date.'

        <associatedValueReference> ValuationReference </associatedValueReference> [1]
        'A reference to the value associated with this scheduled date.'

     End Choice
  End Group: AssociatedValue.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ScheduledDate">
  <xsd:sequence>
    <xsd:group ref=" AdjustedAndOrUnadjustedDate.model "/>
    <xsd:element name="type" type=" ScheduledDateType "/>
    <xsd:element name="assetReference" type=" AnyAssetReference " minOccurs="0"/>
    <xsd:group ref=" AssociatedValue.model " minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ScheduledDates

[Table of contents]

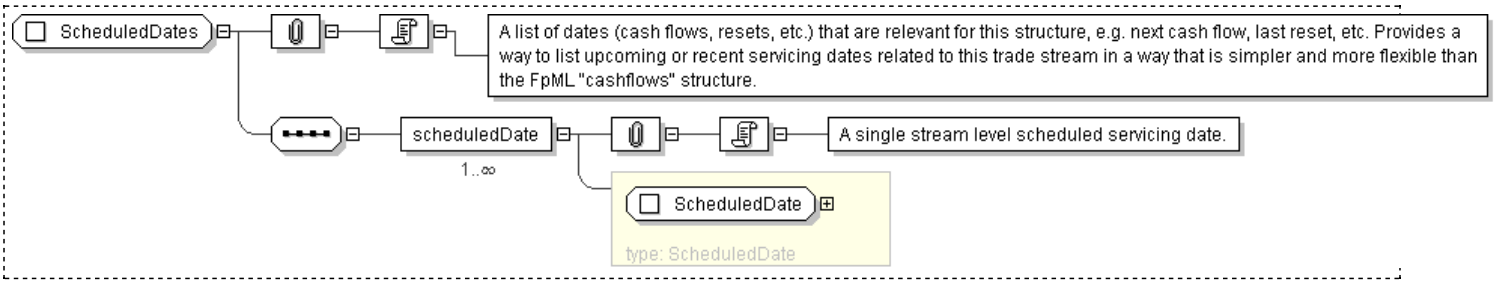
Super-types:	None
Sub-types:	None

Name	ScheduledDates
Abstract	no
Documentation	A list of dates (cash flows, resets, etc.) that are relevant for this structure, e.g. next cash flow, last reset, etc. Provides a way to list upcoming or recent servicing dates related to this trade stream in a way that is simpler and more flexible than the FpML "cashflows" structure.

XML Instance Representation

```
<...>
  <scheduledDate> ScheduledDate </scheduledDate> [1..*]
  'A single stream level scheduled servicing date.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ScheduledDates">
  <xsd:sequence>
    <xsd:element name="scheduledDate" type=" ScheduledDate " maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ScheduledDateType

[Table of contents]

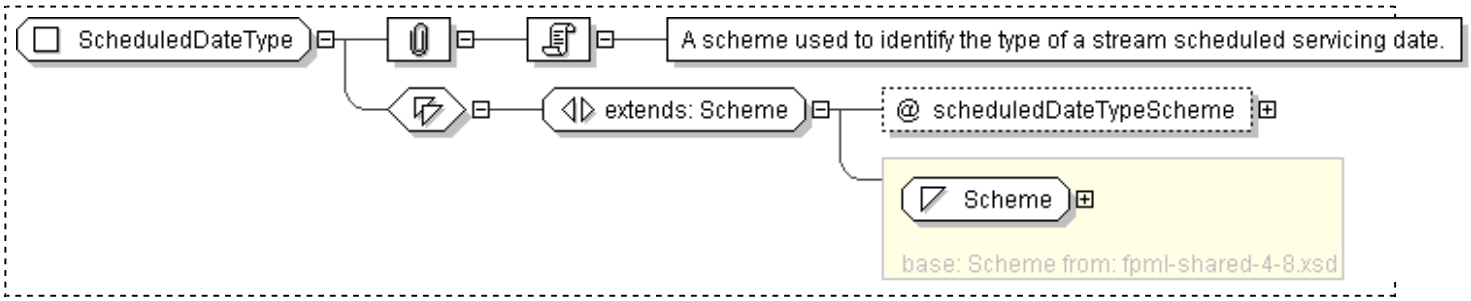
Super-types:	Scheme < ScheduledDateType (by extension)
Sub-types:	None

Name	ScheduledDateType
Used by (from the same schema document)	Complex Type ScheduledDate
Abstract	no
Documentation	A scheme used to identify the type of a stream scheduled servicing date.

XML Instance Representation

```
<...
 scheduledDateTypeScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ScheduledDateType">
   <xsd:simpleContent>
      <xsd:extension base=" Scheme ">
         <xsd:attribute name="scheduledDateTypeScheme" type=" xsd:anyURI "
         default="http://www.fpml.org/coding-scheme/scheduled-date-type"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Sensitivity

[Table of contents]

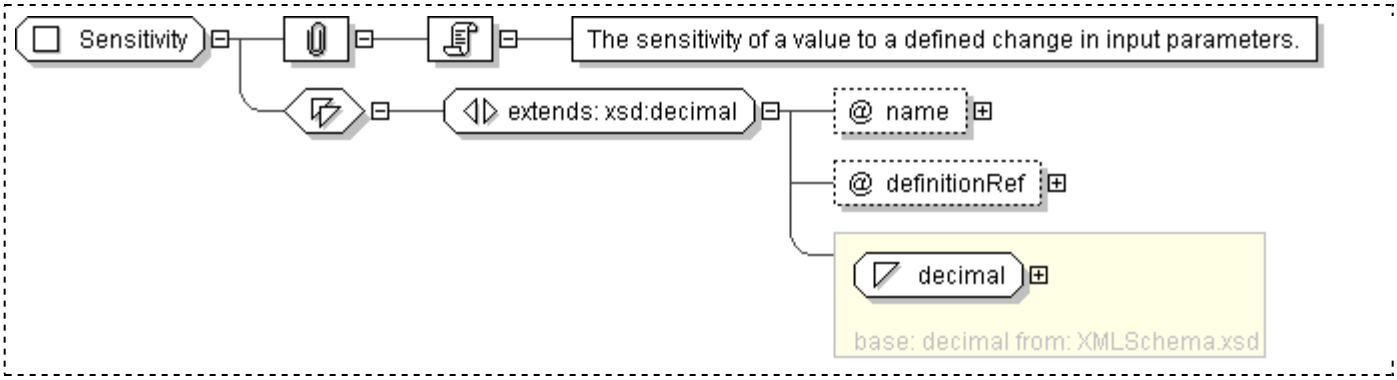
Super-types:	xsd:decimal < Sensitivity (by extension)
Sub-types:	None

Name	Sensitivity
Used by (from the same schema document)	Complex Type SensitivitySet
Abstract	no
Documentation	The sensitivity of a value to a defined change in input parameters.

XML Instance Representation

```
<...
 name=" xsd:normalizedString [0..1]
'A optional name for this sensitivity. This is primarily intended for display
purposes.'

 "
 definitionRef=" xsd:IDREF [0..1]
'A optional (but normally supplied) reference to the definition of this
sensitivity.'

 ">
   xsd:decimal
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Sensitivity">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:decimal ">
         <xsd:attribute name="name" type=" xsd:normalizedString "/>
         <xsd:attribute name="definitionRef" type=" xsd:IDREF "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: SensitivitySet

[Table of contents]

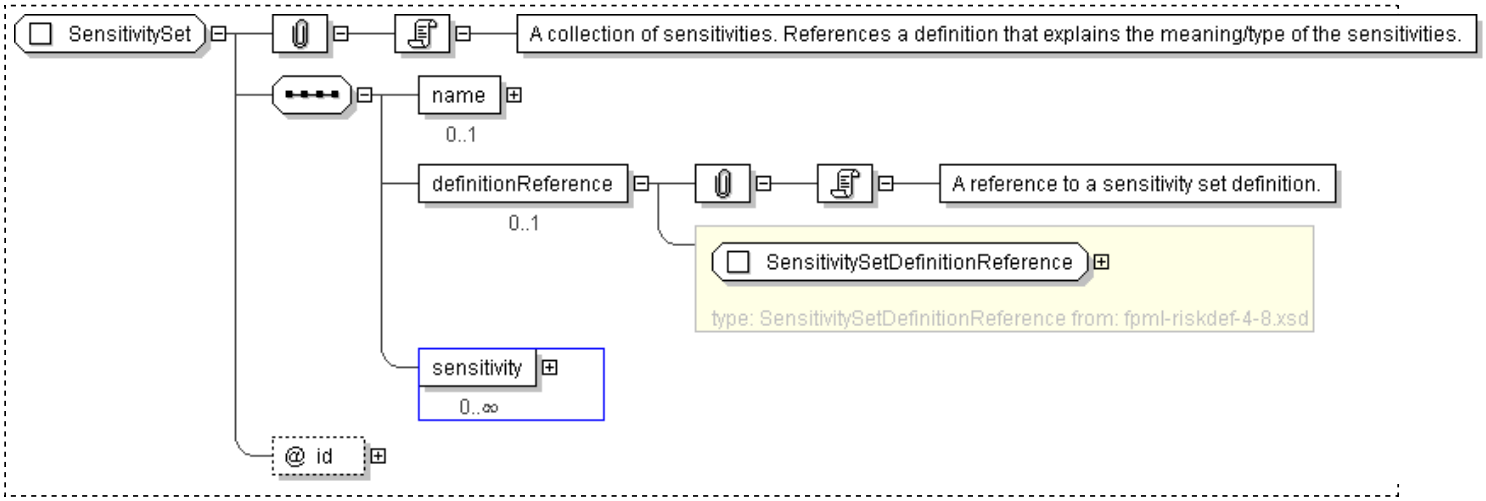
Super-types:	None
Sub-types:	None

Name	SensitivitySet
Used by (from the same schema document)	Complex Type Quotation
Abstract	no
Documentation	A collection of sensitivities. References a definition that explains the meaning/type of the sensitivities.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   <definitionReference> SensitivitySetDefinitionReference </definitionReference> [0..1]
    'A reference to a sensitivity set definition.'

   <sensitivity> Sensitivity </sensitivity> [0..*]
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="SensitivitySet">
   <xsd:sequence>
      <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
      <xsd:element name="definitionReference" type=" SensitivitySetDefinitionReference "
      minOccurs="0"/>
      <xsd:element name="sensitivity" type=" Sensitivity " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: Valuations

[Table of contents]

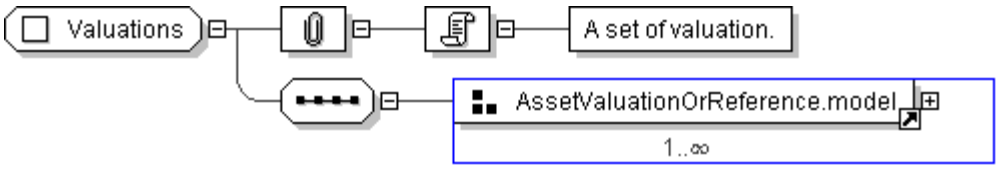
Super-types:	None
Sub-types:	None

Name	Valuations
Abstract	no
Documentation	A set of valuation.

XML Instance Representation

```
<...>
   Start Group: AssetValuationOrReference.model [1..*]
      Start Choice [1]
         <valuation> AssetValuation </valuation> [1]
         ''

         <valuationReference> ValuationReference </valuationReference> [1]
         'A reference to a quotation'

      End Choice
   End Group: AssetValuationOrReference.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Valuations">
   <xsd:sequence>
      <xsd:group ref=" AssetValuationOrReference.model " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationSet

[Table of contents]

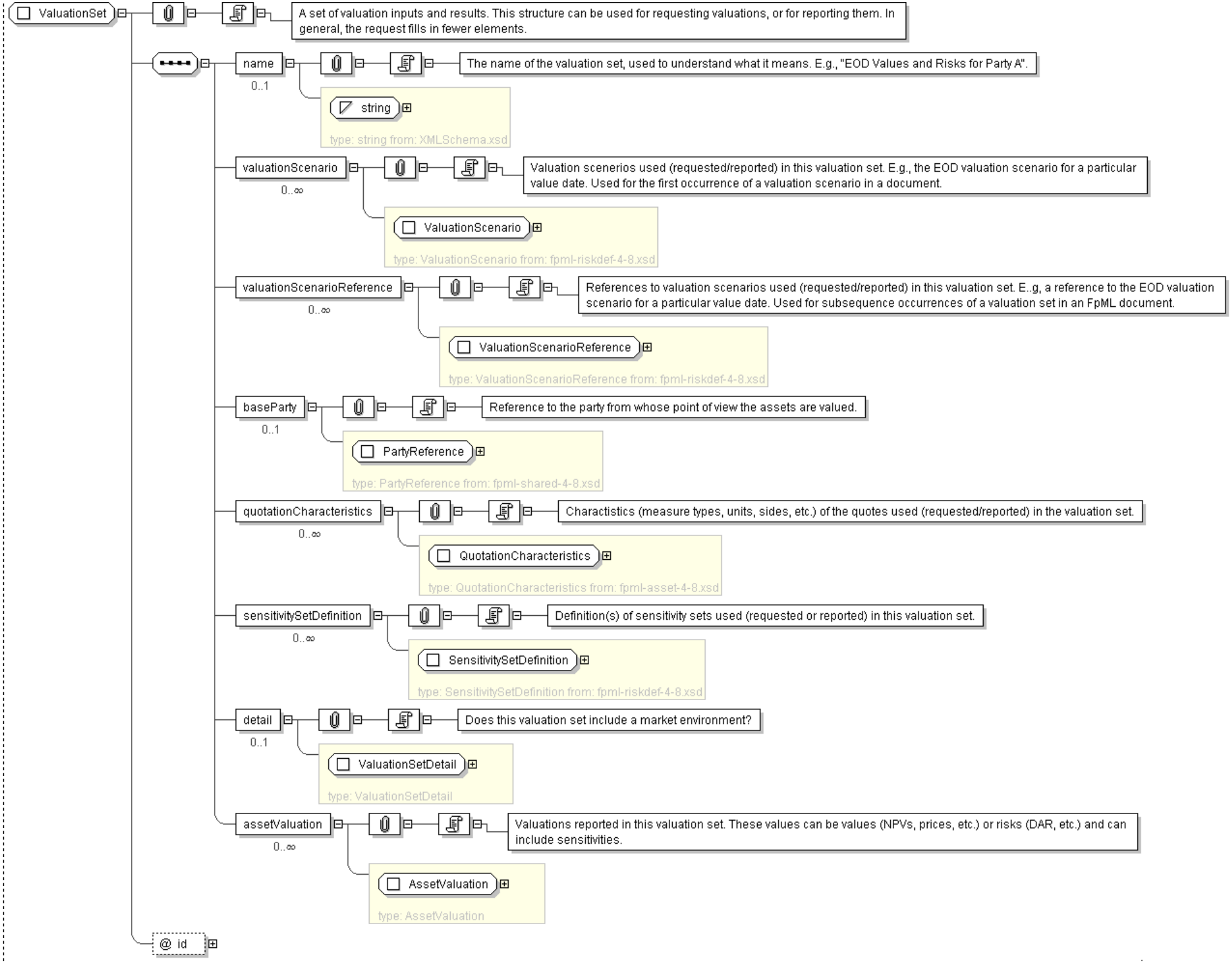
Super-types:	None
Sub-types:	None

Name	ValuationSet
Used by (from the same schema document)	Element valuationSet
Abstract	no
Documentation	A set of valuation inputs and results. This structure can be used for requesting valuations, or for reporting them. In general, the request fills in fewer elements.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <name> xsd:string </name> [0..1]
   'The name of the valuation set, used to understand what it means. E.g., \"EOD Values and Risks for Party A\".'

   <valuationScenario> ValuationScenario </valuationScenario> [0..*]
   'Valuation scenerios used (requested/reported) in this valuation set. E.g., the EOD valuation scenario for a particular value date. Used for
   the first occurrence of a valuation scenario in a document.'

   <valuationScenarioReference> ValuationScenarioReference </valuationScenarioReference> [0..*]
   'References to valuation scenarios used (requested/reported) in this valuation set. E..g, a reference to the EOD valuation scenario for a
   particular value date. Used for subsequence occurrences of a valuation set in an FpML document.'

   <baseParty> PartyReference </baseParty> [0..1]
   'Reference to the party from whose point of view the assets are valued.'

   <quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..*]
   'Charactistics (measure types, units, sides, etc.) of the quotes used (requested/reported) in the valuation set.'

   <sensitivitySetDefinition> SensitivitySetDefinition </sensitivitySetDefinition> [0..*]
   'Definition(s) of sensitivity sets used (requested or reported) in this valuation set.'

   <detail> ValuationSetDetail </detail> [0..1]
   'Does this valuation set include a market environment?'

   <assetValuation> AssetValuation </assetValuation> [0..*]
   'Valuations reported in this valuation set. These values can be values (NPVs, prices, etc.) or risks (DAR, etc.) and can include
   sensitivities.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationSet">
   <xsd:sequence>
      <xsd:element name="name" type=" xsd:string " minOccurs="0"/>
      <xsd:element name="valuationScenario" type=" ValuationScenario " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="valuationScenarioReference" type=" ValuationScenarioReference " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="baseParty" type=" PartyReference " minOccurs="0"/>
      <xsd:element name="quotationCharacteristics" type=" QuotationCharacteristics " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="sensitivitySetDefinition" type=" SensitivitySetDefinition " minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="detail" type=" ValuationSetDetail " minOccurs="0"/>
      <xsd:element name="assetValuation" type=" AssetValuation " minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: ValuationSetDetail

[Table of contents]

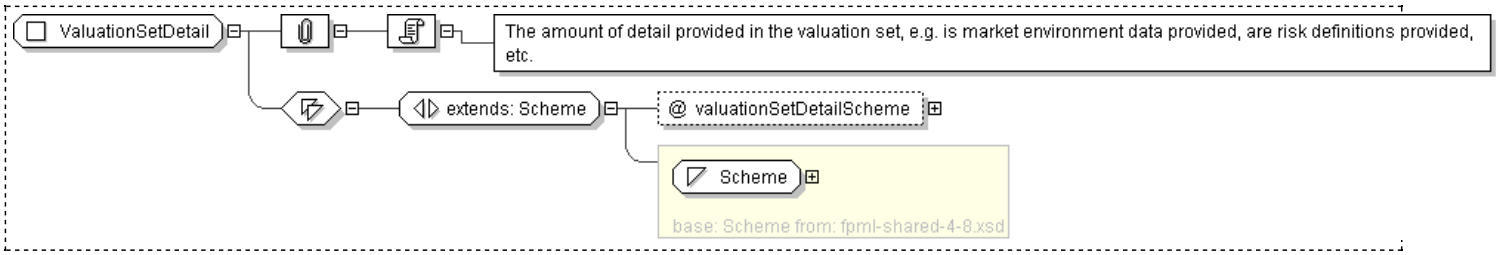
Super-types:	Scheme < ValuationSetDetail (by extension)
Sub-types:	None

Name	ValuationSetDetail
Used by (from the same schema document)	Complex Type ValuationSet
Abstract	no
Documentation	The amount of detail provided in the valuation set, e.g. is market environment data provided, are risk definitions provided, etc.

XML Instance Representation

```
<...
 valuationSetDetailScheme=" xsd:anyURI [0..1]">
   Scheme
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ValuationSetDetail">
  <xsd:simpleContent>
    <xsd:extension base=" Scheme ">
      <xsd:attribute name="valuationSetDetailScheme" type=" xsd:anyURI "/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: varianceOptionTransactionSupplement
  - Element: varianceSwap
  - Element: varianceSwapTransactionSupplement
- Global Definitions
  - Complex Type: VarianceAmount
  - Complex Type: VarianceLeg
  - Complex Type: VarianceOptionTransactionSupplement
  - Complex Type: VarianceSwap
  - Complex Type: VarianceSwapTransactionSupplement
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	http://www.fpml.org/2010/FpML-4-8
Version	$Revision: 2527 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ◦ fpml-eqd-4-8.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.fpml.org/2010/FpML-4-8
ecore	http://www.eclipse.org/emf/2002/Ecore
xml	http://www.w3.org/XML/1998/namespace
fpml-annotation	http://www.fpml.org/annotation
xsd	http://www.w3.org/2001/XMLSchema
fpml	http://www.fpml.org/2010/FpML-4-8

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML"
targetNamespace="http://www.fpml.org/2010/FpML-4-8" version="$Revision: 2527 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xsd:include schemaLocation="fpml-eqd-4-8.xsd"/>
  ...
</xsd:schema>
```

top

Legend

Complex Type: AusAddress

Schema Component Type — Schema Component Name

Super-types: Address < AusAddress (by extension)

Sub-types:

- QLDAddress (by restriction)

If this schema component is a type definition, its type hierarchy is shown in a gray-bordered box.

Name	AusAddress
Abstract	no

The table above displays the properties of this schema component.

XML Instance Representation

```
<... country="Australia" >
   <unitNo> string </unitNo> [0..1]
   <houseNo> string </houseNo> [1]
   <street> string </street> [1]
   Start Choice [1]
      <city> string </city> [1]
      <town> string </town> [1]
   End Choice
   <state> AusStates </state> [1]
   <postcode> string <<pattern = [1-9][0-9]{3}>> </postcode> [1]
</...>
```

The XML Instance Representation table above shows the schema component's content as an XML instance.

- The minimum and maximum occurrence of elements and attributes are provided in square brackets, e.g. [0..1].
- Model group information are shown in gray, e.g. Start Choice ... End Choice.
- For type derivations, the elements and attributes that have been added to or changed from the base type's content are shown in **bold**.
- If an element/attribute has a fixed value, the fixed value is shown in green, e.g. country="Australia".
- Otherwise, the type of the element/attribute is displayed.
  - If the element/attribute's type is in the schema, a link is provided to it.
  - For local simple type definitions, the constraints are displayed in angle brackets, e.g. <<*pattern* = [1-9][0-9]{3}>>.

Schema Component Representation

```
<complexType name="AusAddress">
   <complexContent>
      <extension base=" Address ">
         <sequence>
            <element name="state" type=" AusStates "/>
            <element name="postcode">
               <simpleType>
                  <restriction base=" string ">
                     <pattern value="[1-9][0-9]{3}"/>
                  </restriction>
               </simpleType>
            </element>
         </sequence>
         <attribute name="country" type=" string " fixed="Australia"/>
```

```
      </extension>
    </complexContent>
  </complexType>
```

The Schema Component Representation table above displays the underlying XML representation of the schema component. (Annotations are not shown.)

top

Glossary

Abstract (Applies to complex type definitions and element declarations). An abstract element or complex type cannot used to validate an element instance. If there is a reference to an abstract element, only element declarations that can substitute the abstract element can be used to validate the instance. For references to abstract type definitions, only derived types can be used.

All Model Group Child elements can be provided *in any order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-all.

Choice Model Group *Only one* from the list of child elements and model groups can be provided in instances. See: http://www.w3.org/TR/xmlschema-1/#element-choice.

Collapse Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32). Then, collapse contiguous sequences of space characters into single space character, and remove leading and trailing space characters.

Disallowed Substitutions (Applies to element declarations). If *substitution* is specified, then substitution group members cannot be used in place of the given element declaration to validate element instances. If *derivation methods*, e.g. extension, restriction, are specified, then the given element declaration will not validate element instances that have types derived from the element declaration's type using the specified derivation methods. Normally, element instances can override their declaration's type by specifying an `xsi:type` attribute.

Key Constraint Like Uniqueness Constraint, but additionally requires that the specified value(s) must be provided. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Key Reference Constraint Ensures that the specified value(s) must match value(s) from a Key Constraint or Uniqueness Constraint. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

Model Group Groups together element content, specifying the order in which the element content can occur and the number of times the group of element content may be repeated. See: http://www.w3.org/TR/xmlschema-1/#Model_Groups.

Nillable (Applies to element declarations). If an element declaration is nillable, instances can use the `xsi:nil` attribute. The `xsi:nil` attribute is the boolean attribute, *nil*, from the *http://www.w3.org/2001/XMLSchema-instance* namespace. If an element instance has an `xsi:nil` attribute set to true, it can be left empty, even though its element declaration may have required content.

Notation A notation is used to identify the format of a piece of data. Values of elements and attributes that are of type, NOTATION, must come from the names of declared notations. See: http://www.w3.org/TR/xmlschema-1/#cNotation_Declarations.

Preserve Whitespace Policy Preserve whitespaces exactly as they appear in instances.

Prohibited Derivations (Applies to type definitions). Derivation methods that cannot be used to create sub-types from a given type definition.

Prohibited Substitutions (Applies to complex type definitions). Prevents sub-types that have been derived using the specified derivation methods from validating element instances in place of the given type definition.

Replace Whitespace Policy Replace tab, line feed, and carriage return characters with space character (Unicode character 32).

Sequence Model Group Child elements and model groups must be provided *in the specified order* in instances. See: http://www.w3.org/TR/xmlschema-1/#element-sequence.

Substitution Group Elements that are *members* of a substitution group can be used wherever the *head* element of the substitution group is referenced.

Substitution Group Exclusions (Applies to element declarations). Prohibits element declarations from nominating themselves as being able to substitute a given element declaration, if they have types that are derived from the original element's type using the specified derivation methods.

Target Namespace The target namespace identifies the namespace that components in this schema belongs to. If no target namespace is provided, then the schema components do not belong to any namespace.

Uniqueness Constraint Ensures uniqueness of an element/attribute value, or a combination of values, within a specified scope. See: http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions.

top

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: varianceOptionTransactionSupplement

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	varianceOptionTransactionSupplement
Type	VarianceOptionTransactionSupplement
Nillable	no
Abstract	no
Documentation	Specifies the structure of a variance option.

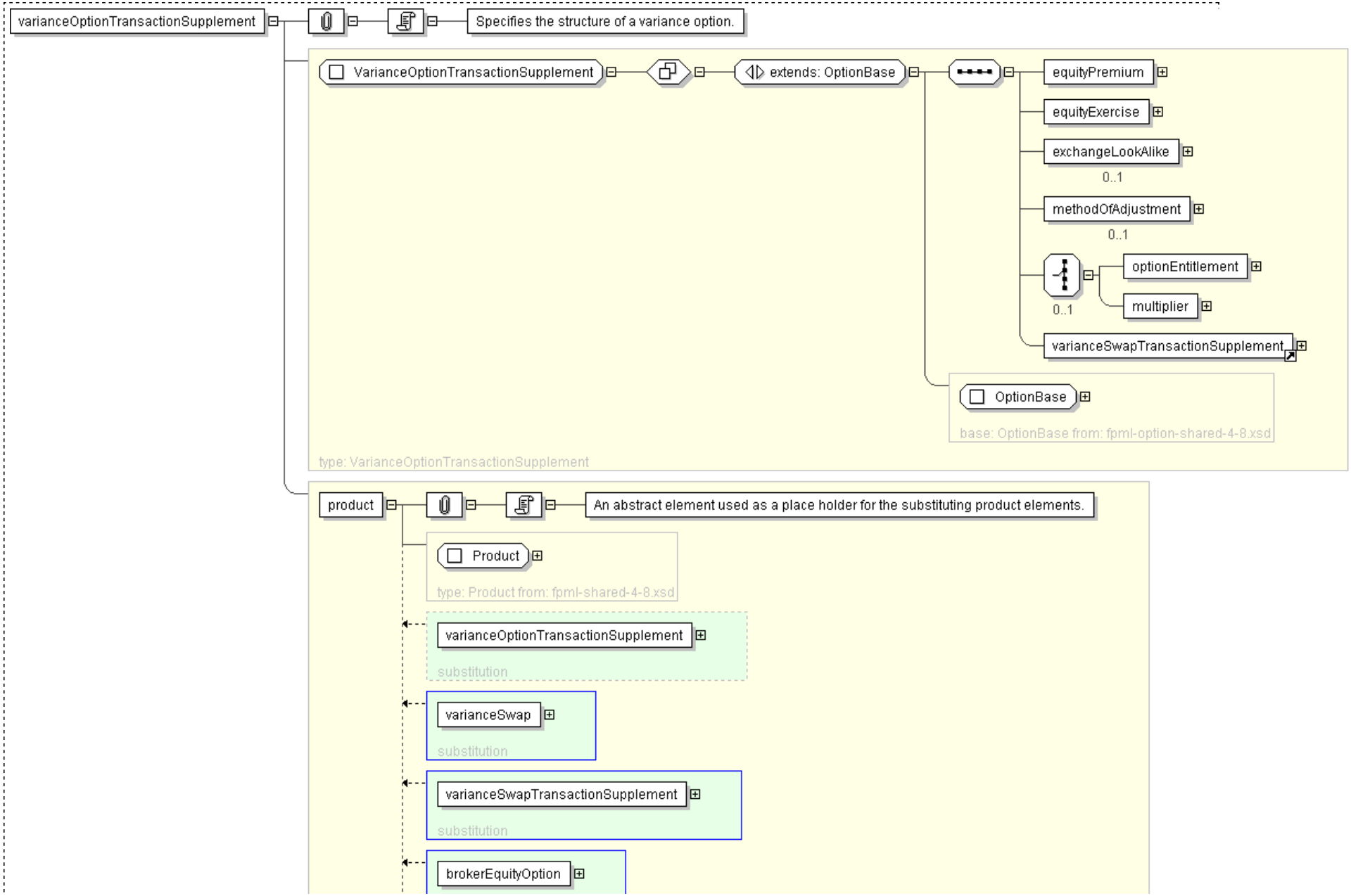
Logical Diagram

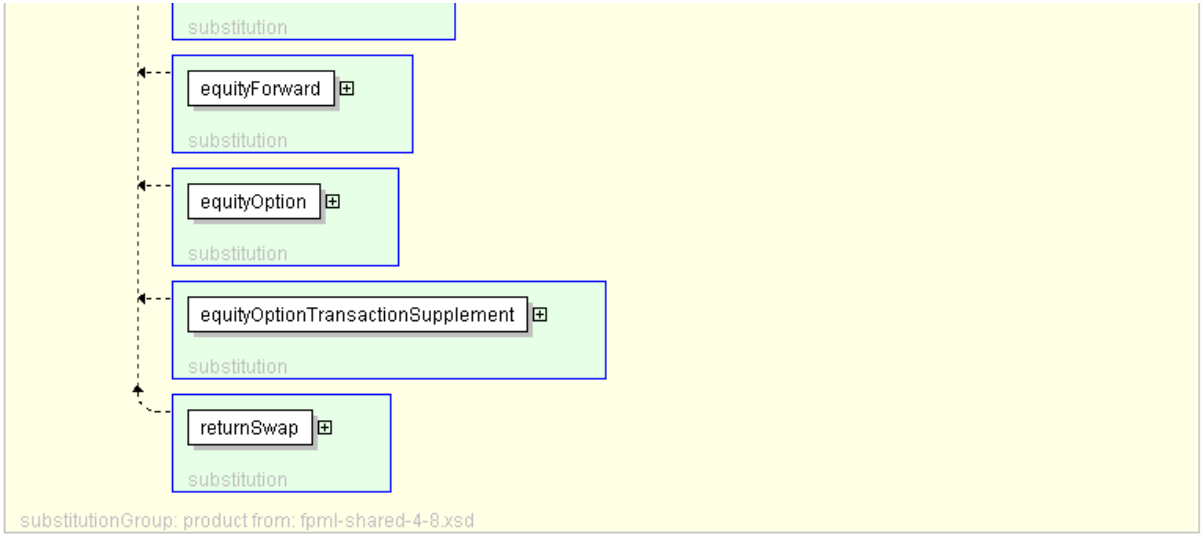
XML Instance Representation

```
<varianceOptionTransactionSupplement
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note
   that the domain values for this element are not strictly an enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000
   ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in
   return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate
   payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default option type, while payer/receiver indicator is
   used for options index credit default swaps, consistently with the industry practice. Straddle is used for the case of straddle
   strategy, that combine a call and a put with the same strike.'

   <equityPremium> EquityPremium </equityPremium> [1]
   'The variance option premium payable by the buyer to the seller.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued and how it is settled.'

   <exchangeLookAlike> xsd:boolean </exchangeLookAlike> [0..1]
   'For a share option transaction, a flag used to indicate whether the transaction is to be treated as an \'exchange look-alike\'.
   This designation has significance for how share adjustments (arising from corporate actions) will be determined for the
   transaction. For an \'exchange look-alike\' transaction the relevant share adjustments will follow that for a corresponding
   designated contract listed on the related exchange (referred to as Options Exchange Adjustment (ISDA defined term), otherwise the
   share adjustments will be determined by the calculation agent (referred to as Calculation Agent Adjustment (ISDA defined term)).'

   <methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [0..1]
   'Defines how adjustments will be made to the contract should one or more of the extraordinary events occur.'

   Start Choice [0..1]
     <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
```

```
      'The number of shares per option comprised in the option transaction supplement.'

      <multiplier> PositiveDecimal </multiplier> [1]
      'Specifies the contract multiplier that can be associated with an index option.'

   End Choice
   <varianceSwapTransactionSupplement> ... </varianceSwapTransactionSupplement> [1]
   'The variance swap details.'

</varianceOptionTransactionSupplement>
```

Diagram





Schema Component Representation

```
<xsd:element name="varianceOptionTransactionSupplement" type=" VarianceOptionTransactionSupplement " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

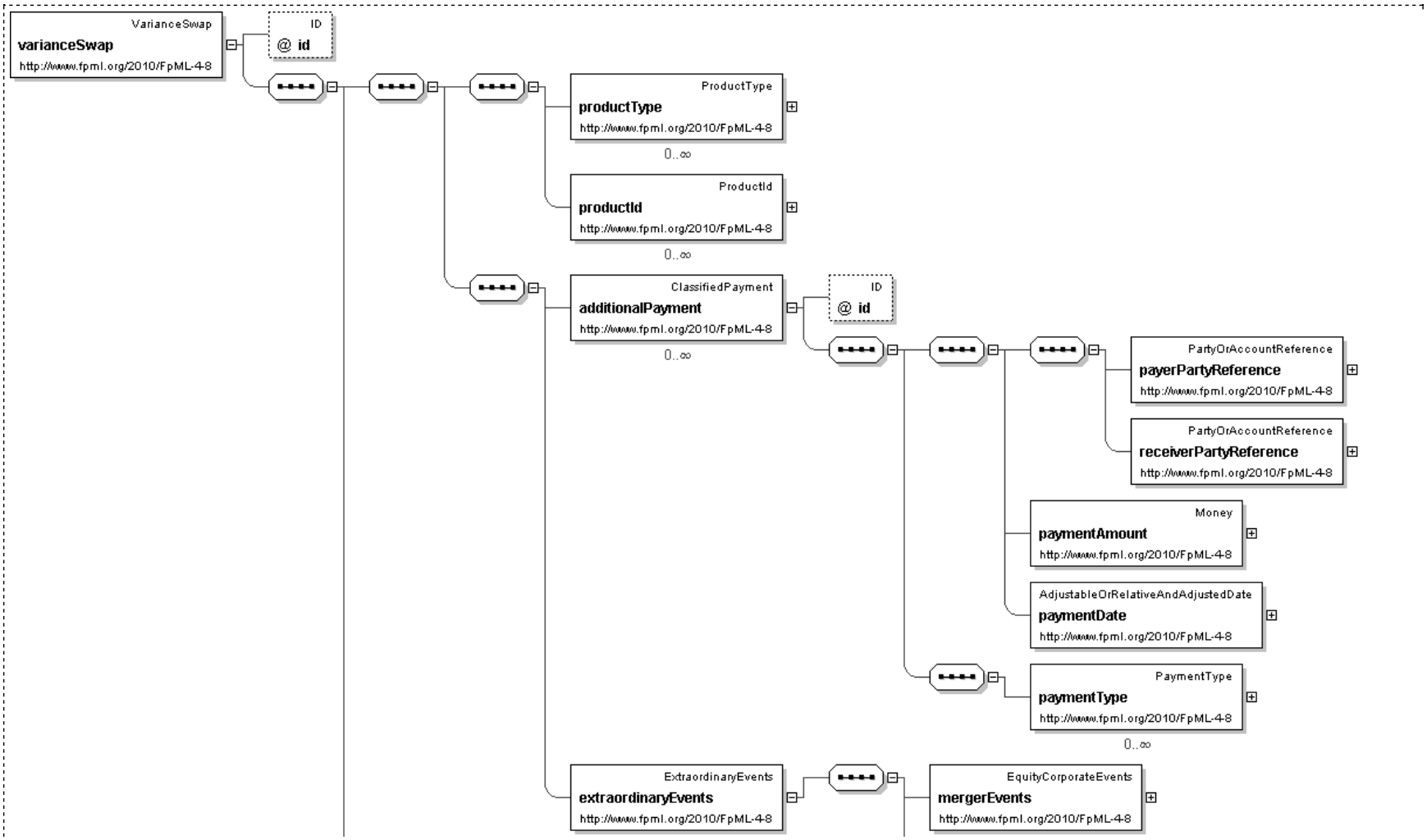
Element: varianceSwap

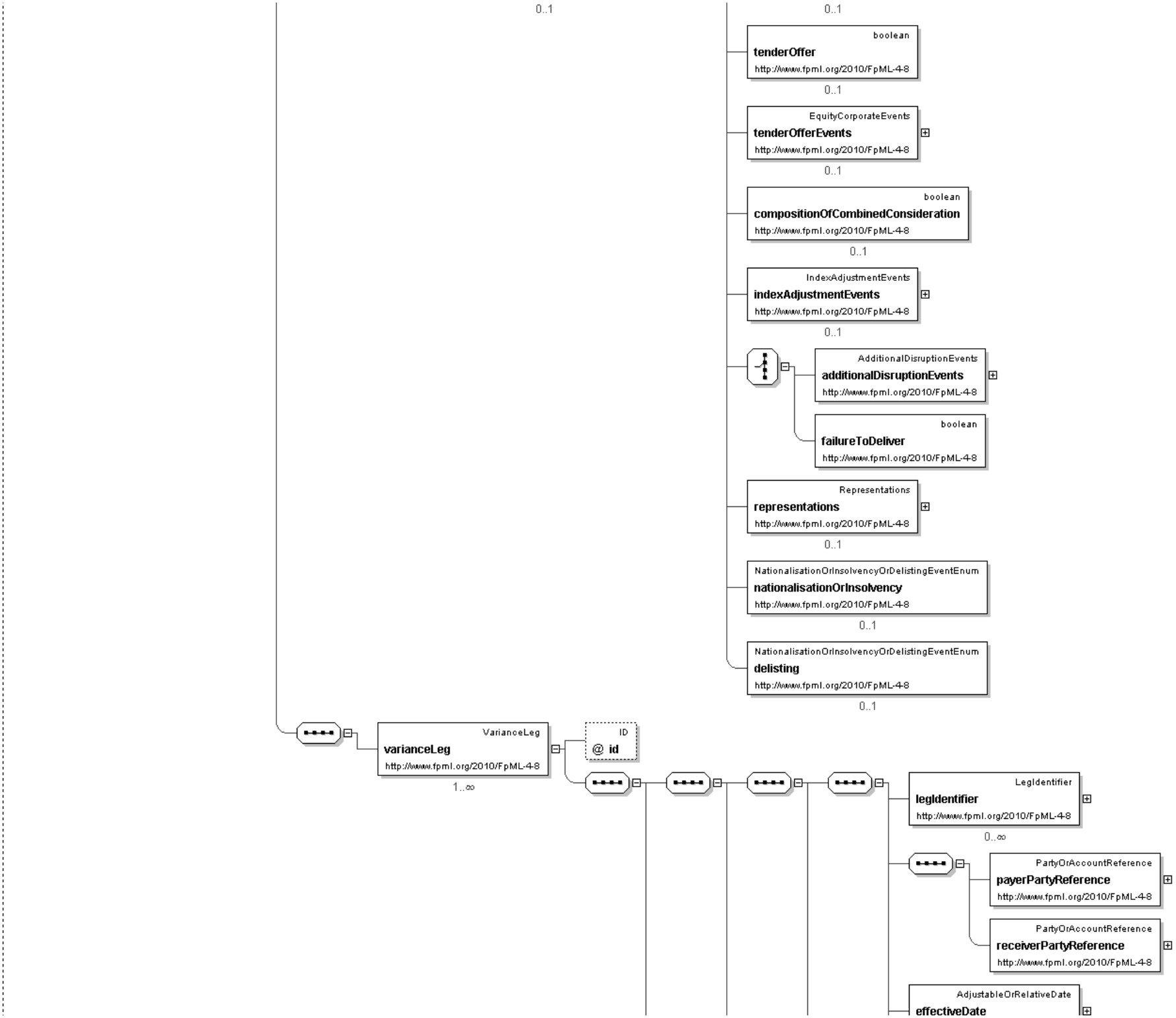
[Table of contents]

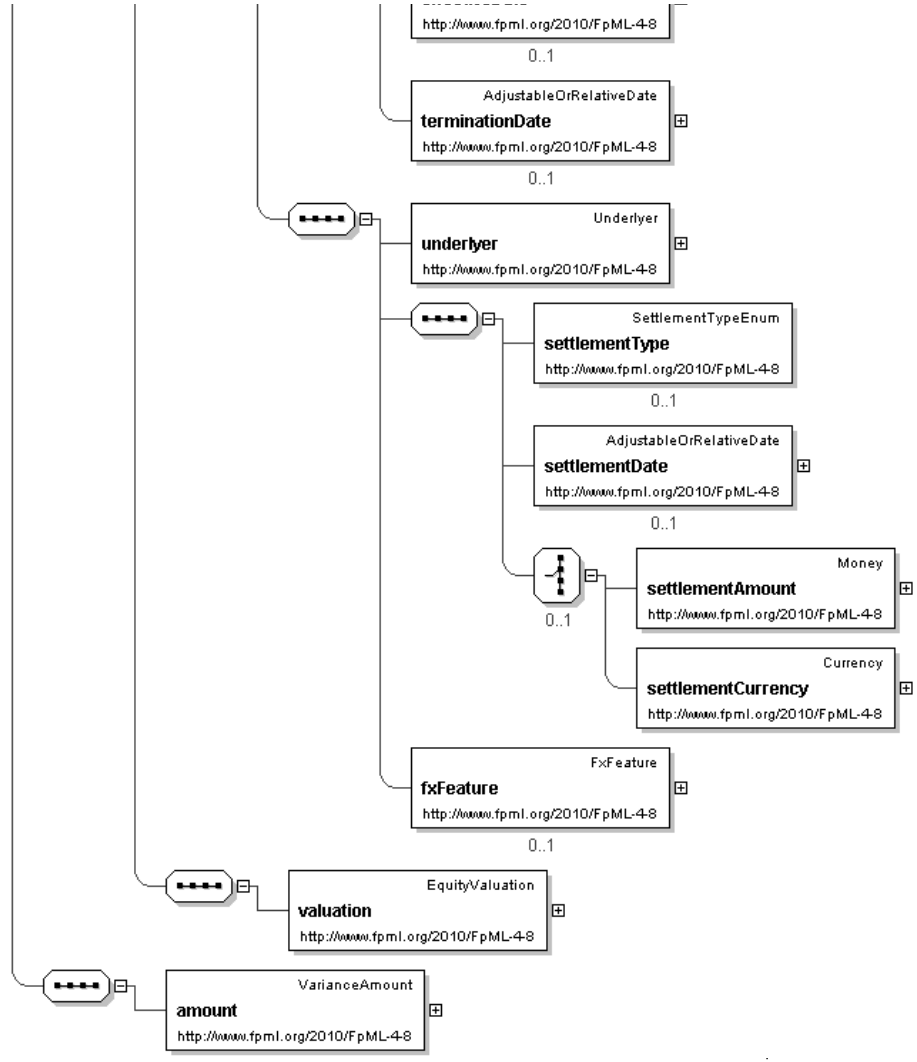
- *This element can be used wherever the following element is referenced:*
  - product

Name	varianceSwap
Type	VarianceSwap
Nillable	no
Abstract	no
Documentation	Specifies the structure of a variance swap.

Logical Diagram



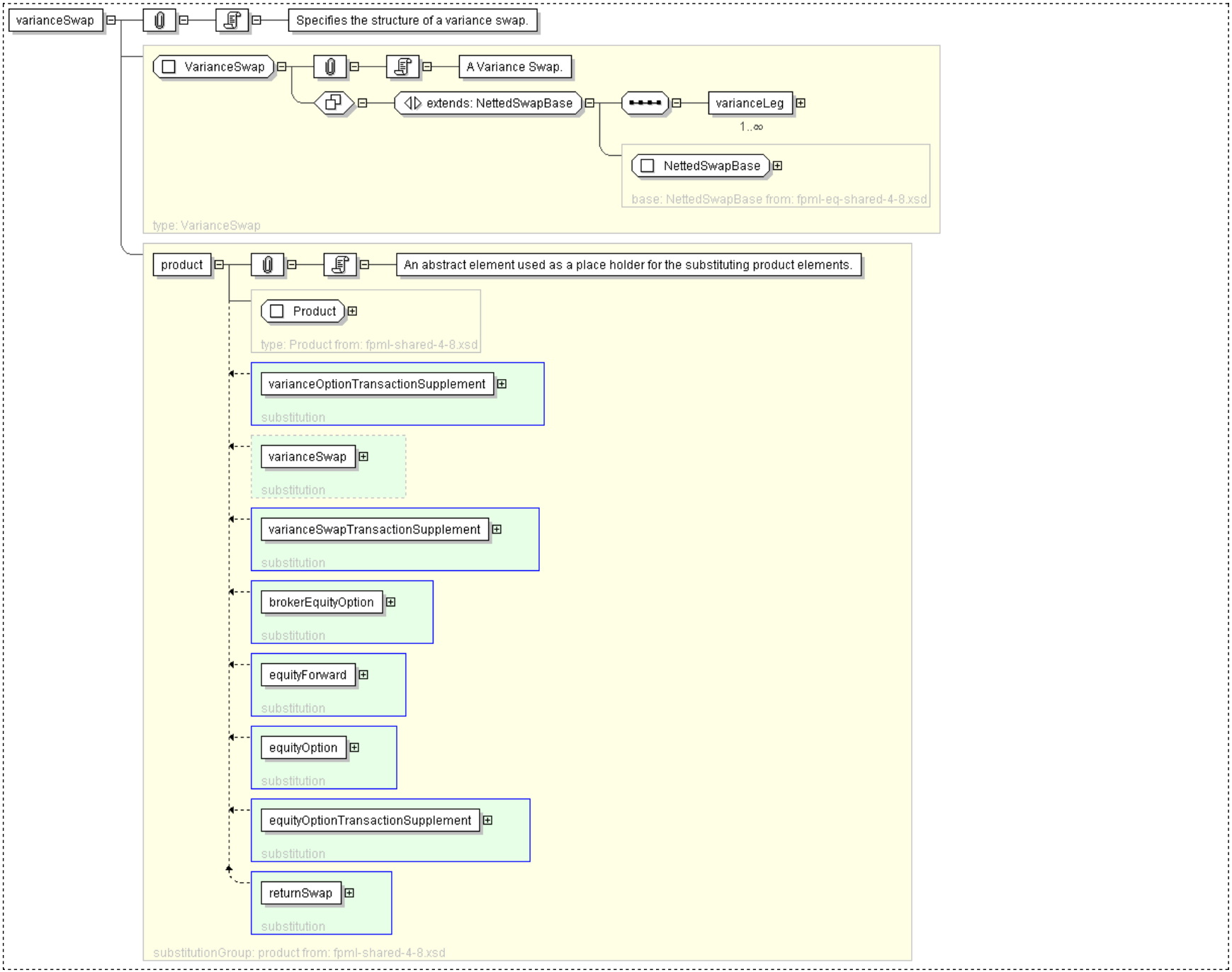

0..1
0..1
boolean
tenderOffer
http://www.fpml.org/2010/FpML-4-8
0..1
EquityCorporateEvents
tenderOfferEvents
http://www.fpml.org/2010/FpML-4-8
0..1
boolean
compositionOfCombinedConsideration
http://www.fpml.org/2010/FpML-4-8
0..1
IndexAdjustmentEvents
indexAdjustmentEvents
http://www.fpml.org/2010/FpML-4-8
0..1
AdditionalDisruptionEvents
additionalDisruptionEvents
http://www.fpml.org/2010/FpML-4-8
boolean
failureToDeliver
http://www.fpml.org/2010/FpML-4-8
Representations
representations
http://www.fpml.org/2010/FpML-4-8
0..1
NationalisationOrInsolvencyOrDelistingEventEnum
nationalisationOrInsolvency
http://www.fpml.org/2010/FpML-4-8
0..1
NationalisationOrInsolvencyOrDelistingEventEnum
delisting
http://www.fpml.org/2010/FpML-4-8
0..1
VarianceLeg
varianceLeg
http://www.fpml.org/2010/FpML-4-8
1..∞
ID
@ id
LegIdentifier
legIdentifier
http://www.fpml.org/2010/FpML-4-8
0..∞
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2010/FpML-4-8
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2010/FpML-4-8
AdjustableOrRelativeDate
effectiveDate



XML Instance Representation

```
<varianceSwap
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain
    values for this element are not strictly an enumerated list.'

   <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
    'Specifies additional payment(s) between the principal parties to the netted swap.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
    'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be
    adjusted.'

   <varianceLeg> VarianceLeg </varianceLeg> [1..*]
    'Variance Leg.'

</varianceSwap>
```

Diagram



Schema Component Representation

```
<xsd:element name="varianceSwap" type=" VarianceSwap " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Element: varianceSwapTransactionSupplement

[Table of contents]

- *This element can be used wherever the following element is referenced:*
  - product

Name	varianceSwapTransactionSupplement
Used by (from the same schema document)	Complex Type VarianceOptionTransactionSupplement
Type	VarianceSwapTransactionSupplement
Nillable	no
Abstract	no
Documentation	Specifies the structure of a variance swap transaction supplement.

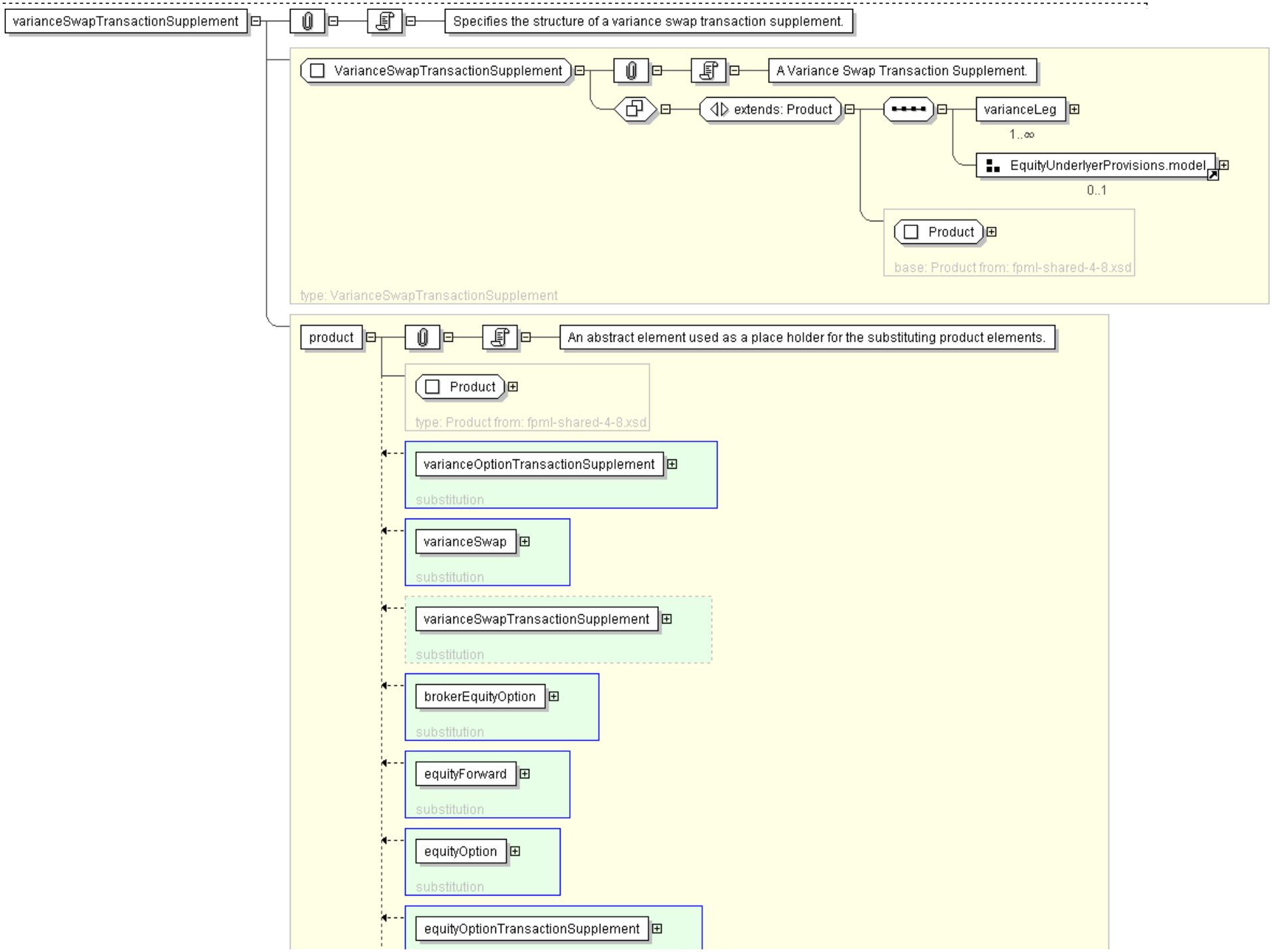
Logical Diagram

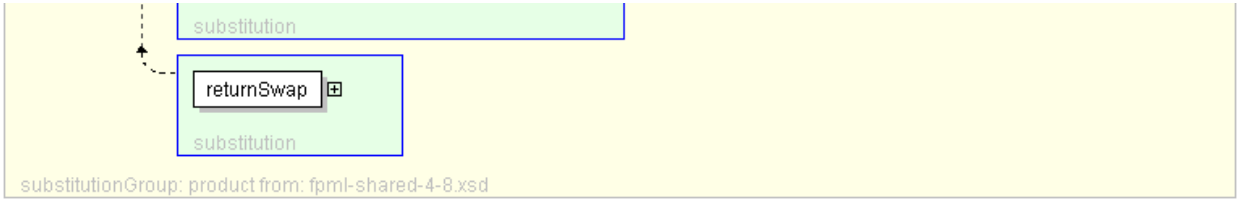
XML Instance Representation

```
<varianceSwapTransactionSupplement
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element.
    Note that the domain values for this element are not strictly an enumerated list.'

   <varianceLeg> VarianceLeg </varianceLeg> [1..*]
    'Variance Leg.'

   Start Group: EquityUnderlyerProvisions.model [0..1]
      Start Group: IndexAnnexFallback.model [0..1]
         Start Choice [1]
            <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
             'For an index option transaction, a flag to indicate whether a relevant Multiple Exchange Index Annex is applicable to
             the transaction. This annex defines additional provisions which are applicable where an index is comprised of component
             securities that are traded on multiple exchanges.'

            <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback> [1]
             'For an index option transaction, a flag to indicate whether a relevant Component Security Index Annex is applicable to
             the transaction.'

         End Choice
      End Group: IndexAnnexFallback.model
      <localJurisdiction> Country </localJurisdiction> [0..1]
       'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall mean
       taxes, duties, and similar charges imposed by the taxing authority of the Local Jurisdiction If this element is not present
       Local Jurisdiction is Not Applicable.'

      <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
       'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to determine local taxes, which shall
       mean taxes, duties and similar charges that would be imposed by the taxing authority of the Country of Underlyer on a
       Hypothetical Broker Dealer assuming the Applicable Hedge Positions are held by its office in the Relevant Jurisdiction. If
       this element is not present Relevant Jurisdiction is Not Applicable.'
```

```
  End Group: EquityUnderlyerProvisions.model
</varianceSwapTransactionSupplement>
```

Diagram





Schema Component Representation

```
<xsd:element name="varianceSwapTransactionSupplement" type=" VarianceSwapTransactionSupplement " substitutionGroup="product"/>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VarianceAmount

[Table of contents]

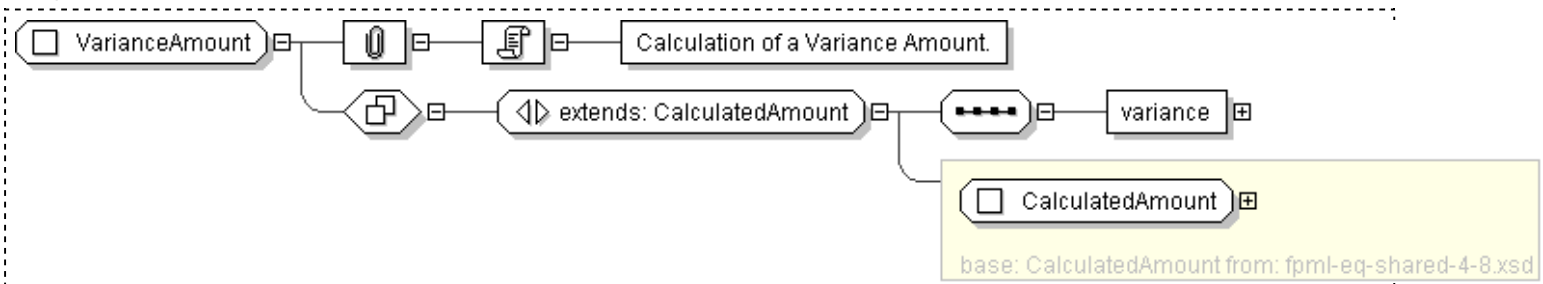
Super-types:	CalculatedAmount < VarianceAmount (by extension)
Sub-types:	None

Name	VarianceAmount
Used by (from the same schema document)	Complex Type VarianceLeg
Abstract	no
Documentation	Calculation of a Variance Amount.

XML Instance Representation

```
<...>
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for
   the purpose of calculating the amount.'

   <observationStartDate> AdjustableOrRelativeDate </observationStartDate> [0..1]
   'The start of the period over which observations are made which are used in the
   calculation Used when the observation start date differs from the trade date such
   as for forward starting swaps.'

   <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
   'If present and true, then options exchange dividends are applicable.'

   <additionalDividends> xsd:boolean </additionalDividends> [0..1]
   'If present and true, then additional dividends are applicable.'

   <allDividends> xsd:boolean </allDividends> [0..1]
   'Represents the European Master Confirmation value of \'All Dividends\' which, when
   applicable, signifies that, for a given Ex-Date, the daily observed Share Price
   for that day is adjusted (reduced) by the cash dividend and/or the cash value of
   any non cash dividend per Share (including Extraordinary Dividends) declared by the
   Issuer.'

   <variance> Variance </variance> [1]
   'Specifies Variance.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VarianceAmount">
   <xsd:complexContent>
      <xsd:extension base=" CalculatedAmount ">
         <xsd:sequence>
            <xsd:element name="variance" type=" Variance "/>
```

```
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VarianceLeg

[Table of contents]

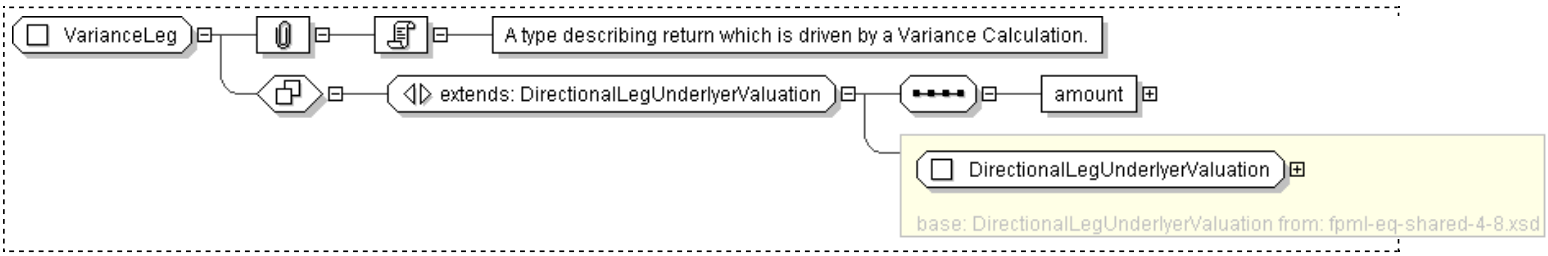
Super-types:	DirectionalLegUnderlyerValuation < VarianceLeg (by extension)
Sub-types:	None

Name	VarianceLeg
Used by (from the same schema document)	Complex Type VarianceSwap , Complex Type VarianceSwapTransactionSupplement
Abstract	no
Documentation	A type describing return which is driven by a Variance Calculation.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <legIdentifier> LegIdentifier </legIdentifier> [0..*]
   'Version aware identification of this leg.'

   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a date specified
   somewhere else in the document (through the relativeDate component), this element will typically
   point to the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
   'Specifies the termination date of this leg of the swap. When defined in relation to a date
   specified somewhere else in the document (through the relativeDate component), this element will
   typically point to the termination date of the other leg of the swap.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <settlementType> SettlementTypeEnum </settlementType> [0..1]
   <settlementDate> AdjustableOrRelativeDate </settlementDate> [0..1]
   Start Group: SettlementAmountOrCurrency.model [0..1]
      Start Choice [1]
         <settlementAmount> Money </settlementAmount> [1]
         'Settlement Amount'

         <settlementCurrency> Currency </settlementCurrency> [1]
         'Settlement Currency for use where the Settlement Amount cannot be known in advance'

      End Choice
   End Group: SettlementAmountOrCurrency.model
   <fxFeature> FxFeature </fxFeature> [0..1]
   'Quanto, Composite, or Cross Currency FX features.'

   <valuation> EquityValuation </valuation> [1]
   'Valuation of the underlyer.'

   <amount> VarianceAmount </amount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date
   relates. Unless otherwise specified, this term has the meaning defined in the ISDA 2002 Equity
   Derivatives Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VarianceLeg">
  <xsd:complexContent>
    <xsd:extension base=" DirectionalLegUnderlyerValuation ">
      <xsd:sequence>
        <xsd:element name="amount" type=" VarianceAmount "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VarianceOptionTransactionSupplement

[Table of contents]

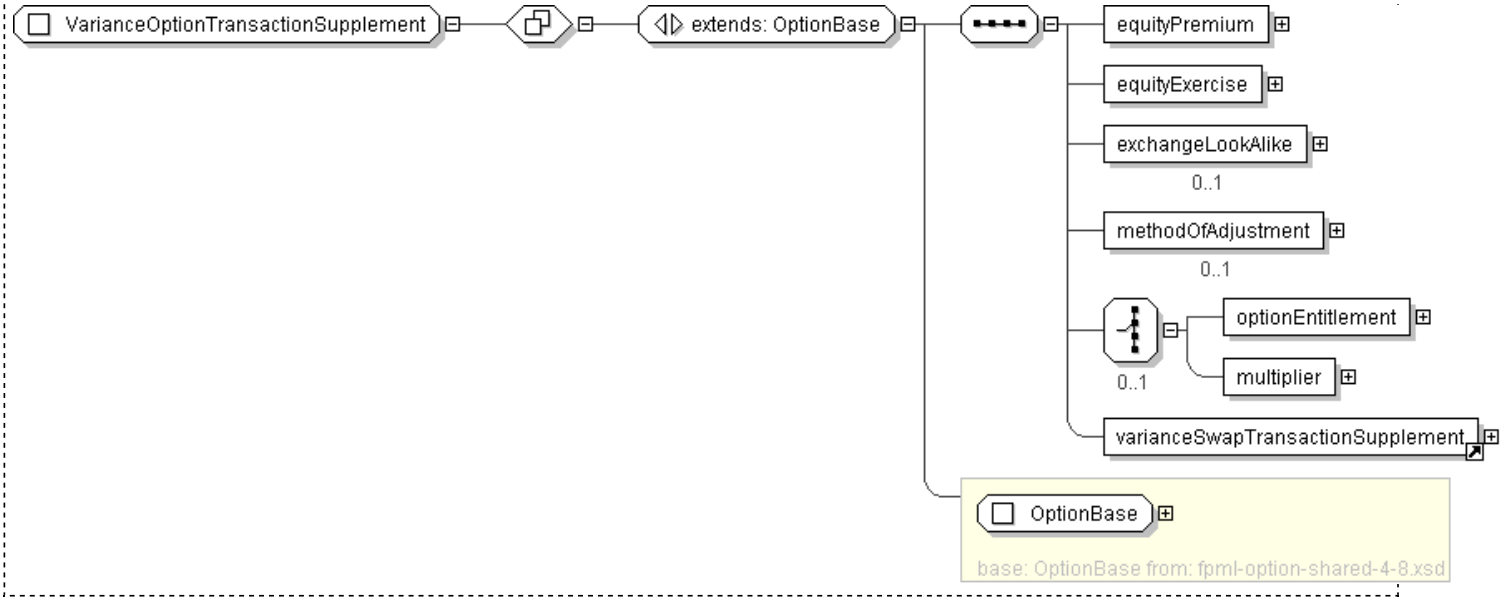
Super-types:	OptionBase < VarianceOptionTransactionSupplement (by extension)
Sub-types:	None

Name	VarianceOptionTransactionSupplement
Used by (from the same schema document)	Element varianceOptionTransactionSupplement
Abstract	no

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using a
   coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain values
   associated with this element. Note that the domain values for this element are not strictly an
   enumerated list.'

   <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
   'A reference to the party that buys this instrument, ie. pays for this instrument and receives
   the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this
   the fixed rate payer.'

   <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
   'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights
   defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions
   Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

   <optionType> OptionTypeEnum </optionType> [1]
   'The type of option transaction. From a usage standpoint, put/call is the default option type,
   while payer/receiver indicator is used for options index credit default swaps, consistently with
   the industry practice. Straddle is used for the case of straddle strategy, that combine a call
   and a put with the same strike.'

   <equityPremium> EquityPremium </equityPremium> [1]
   'The variance option premium payable by the buyer to the seller.'

   <equityExercise> EquityExerciseValuationSettlement </equityExercise> [1]
   'The parameters for defining how the equity option can be exercised, how it is valued and how it
   is settled.'

   <exchangeLookAlike> xsd:boolean </exchangeLookAlike> [0..1]
   'For a share option transaction, a flag used to indicate whether the transaction is to be treated
   as an \'exchange look-alike\'. This designation has significance for how share adjustments
   (arising from corporate actions) will be determined for the transaction. For an \'exchange look-
   alike\' transaction the relevant share adjustments will follow that for a corresponding designated
   contract listed on the related exchange (referred to as Options Exchange Adjustment (ISDA defined
   term), otherwise the share adjustments will be determined by the calculation agent (referred to as
   Calculation Agent Adjustment (ISDA defined term)).'

   <methodOfAdjustment> MethodOfAdjustmentEnum </methodOfAdjustment> [0..1]
   'Defines how adjustments will be made to the contract should one or more of the extraordinary
   events occur.'

   Start Choice [0..1]
      <optionEntitlement> PositiveDecimal </optionEntitlement> [1]
      'The number of shares per option comprised in the option transaction supplement.'

      <multiplier> PositiveDecimal </multiplier> [1]
      'Specifies the contract multiplier that can be associated with an index option.'

   End Choice
   <varianceSwapTransactionSupplement> ... </varianceSwapTransactionSupplement> [1]
   'The variance swap details.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VarianceOptionTransactionSupplement">
  <xsd:complexContent>
    <xsd:extension base=" OptionBase ">
      <xsd:sequence>
        <xsd:element name="equityPremium" type=" EquityPremium "/>
        <xsd:element name="equityExercise" type=" EquityExerciseValuationSettlement "/>
        <xsd:element name="exchangeLookAlike" type=" xsd:boolean " minOccurs="0"/>
        <xsd:element name="methodOfAdjustment" type=" MethodOfAdjustmentEnum " minOccurs="0"/>
        <xsd:choice minOccurs="0">
          <xsd:element name="optionEntitlement" type=" PositiveDecimal "/>
          <xsd:element name="multiplier" type=" PositiveDecimal "/>
        </xsd:choice>
        <xsd:element ref=" varianceSwapTransactionSupplement "/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VarianceSwap

[Table of contents]

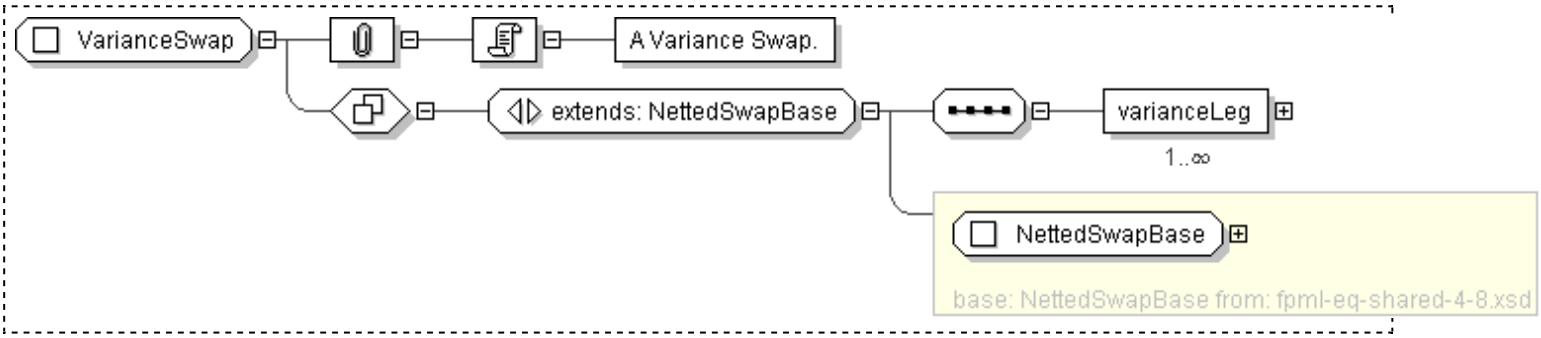
Super-types:	NettedSwapBase < VarianceSwap (by extension)
Sub-types:	None

Name	VarianceSwap
Used by (from the same schema document)	Element varianceSwap
Abstract	no
Documentation	A Variance Swap.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product
   categorization using a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the
   domain values associated with this element. Note that the domain values for this
   element are not strictly an enumerated list.'

   <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
   'Specifies additional payment(s) between the principal parties to the netted
   swap.'

   <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
   'Where the underlying is shares, specifies events affecting the issuer of those
   shares that may require the terms of the transaction to be adjusted.'

   <varianceLeg> VarianceLeg </varianceLeg> [1..*]
   'Variance Leg.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VarianceSwap">
   <xsd:complexContent>
      <xsd:extension base=" NettedSwapBase ">
         <xsd:sequence>
            <xsd:element name="varianceLeg" type=" VarianceLeg " maxOccurs="unbounded"/>
         </xsd:sequence>
```

```
      </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.

XML Schema Documentation

Complex Type: VarianceSwapTransactionSupplement

[Table of contents]

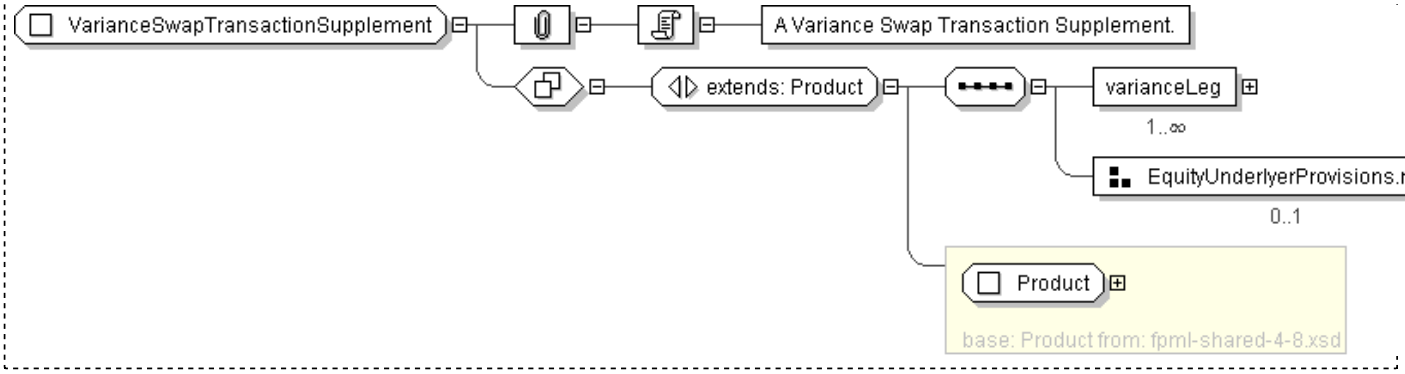
Super-types:	Product < VarianceSwapTransactionSupplement (by extension)
Sub-types:	None

Name	VarianceSwapTransactionSupplement
Used by (from the same schema document)	Element varianceSwapTransactionSupplement
Abstract	no
Documentation	A Variance Swap Transaction Supplement.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using a
    coding scheme.'

   <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain values
    associated with this element. Note that the domain values for this element are not strictly an
    enumerated list.'

   <varianceLeg> VarianceLeg </varianceLeg> [1..*]
    'Variance Leg.'

   Start Group: EquityUnderlyerProvisions.model [0..1]
      Start Group: IndexAnnexFallback.model [0..1]
         Start Choice [1]
            <multipleExchangeIndexAnnexFallback> xsd:boolean </multipleExchangeIndexAnnexFallback> [1]
             'For an index option transaction, a flag to indicate whether a relevant Multiple
             Exchange Index Annex is applicable to the transaction. This annex defines additional
             provisions which are applicable where an index is comprised of component securities
             that are traded on multiple exchanges.'

            <componentSecurityIndexAnnexFallback> xsd:boolean </componentSecurityIndexAnnexFallback>
            [1]
             'For an index option transaction, a flag to indicate whether a relevant Component
             Security Index Annex is applicable to the transaction.'

         End Choice
      End Group: IndexAnnexFallback.model
      <localJurisdiction> Country </localJurisdiction> [0..1]
       'Local Jurisdiction is a term used in the AEJ Master Confirmation, which is used to
       determine local taxes, which shall mean taxes, duties, and similar charges imposed by the
       taxing authority of the Local Jurisdiction If this element is not present Local Jurisdiction
       is Not Applicable.'

      <relevantJurisdiction> Country </relevantJurisdiction> [0..1]
       'Relevent Jurisdiction is a term used in the AEJ Master Confirmation, which is used to
       determine local taxes, which shall mean taxes, duties and similar charges that would be
       imposed by the taxing authority of the Country of Underlyer on a Hypothetical Broker Dealer
       assuming the Applicable Hedge Positions are held by its office in the Relevant
       Jurisdiction. If this element is not present Relevant Jurisdiction is Not Applicable.'

   End Group: EquityUnderlyerProvisions.model
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="VarianceSwapTransactionSupplement">
  <xsd:complexContent>
    <xsd:extension base=" Product ">
      <xsd:sequence>
        <xsd:element name="varianceLeg" type=" VarianceLeg " maxOccurs="unbounded"/>
        <xsd:group ref=" EquityUnderlyerProvisions.model " minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Generated by <oXygen/> XML Editor using a modified version of xs3p that adds schema diagrams and chunking support.