XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: **dividendSwapTransactionSupplement**
  - Element: **dividendSwapTransactionSupplementOption**
- Global Definitions
  - Complex Type: **DividendLeg**
  - Complex Type: **DividendPeriodPayment**
  - Complex Type: **DividendSwapTransactionSupplement**
  - Complex Type: **DividendSwapTransactionSupplementOption**
  - Complex Type: **FixedPaymentAmount**
  - Complex Type: **FixedPaymentLeg**
- Legend
- Glossary

top

Schema Document Properties

Target Namespace	**http://www.fpml.org/2007/FpML-4-4**
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-4-4.xsd
	  ○ fpml-shared-4-4.xsd

Declared Namespaces

Prefix	Namespace
Default namespace	**http://www.fpml.org/2007/FpML-4-4**
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/2007/FpML-4-4**

Schema Component Representation

```
<xsd:schema nsPrefix="fpml" package="org.fpml" documentRoot="FpML" targetNamespace="http://
www.fpml.org/2007/FpML-4-4" version="$Revision: 2527 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-eq-shared-4-4.xsd"/>
   <xsd:include schemaLocation="fpml-shared-4-4.xsd"/>
   ...
</xsd:schema>
```

top

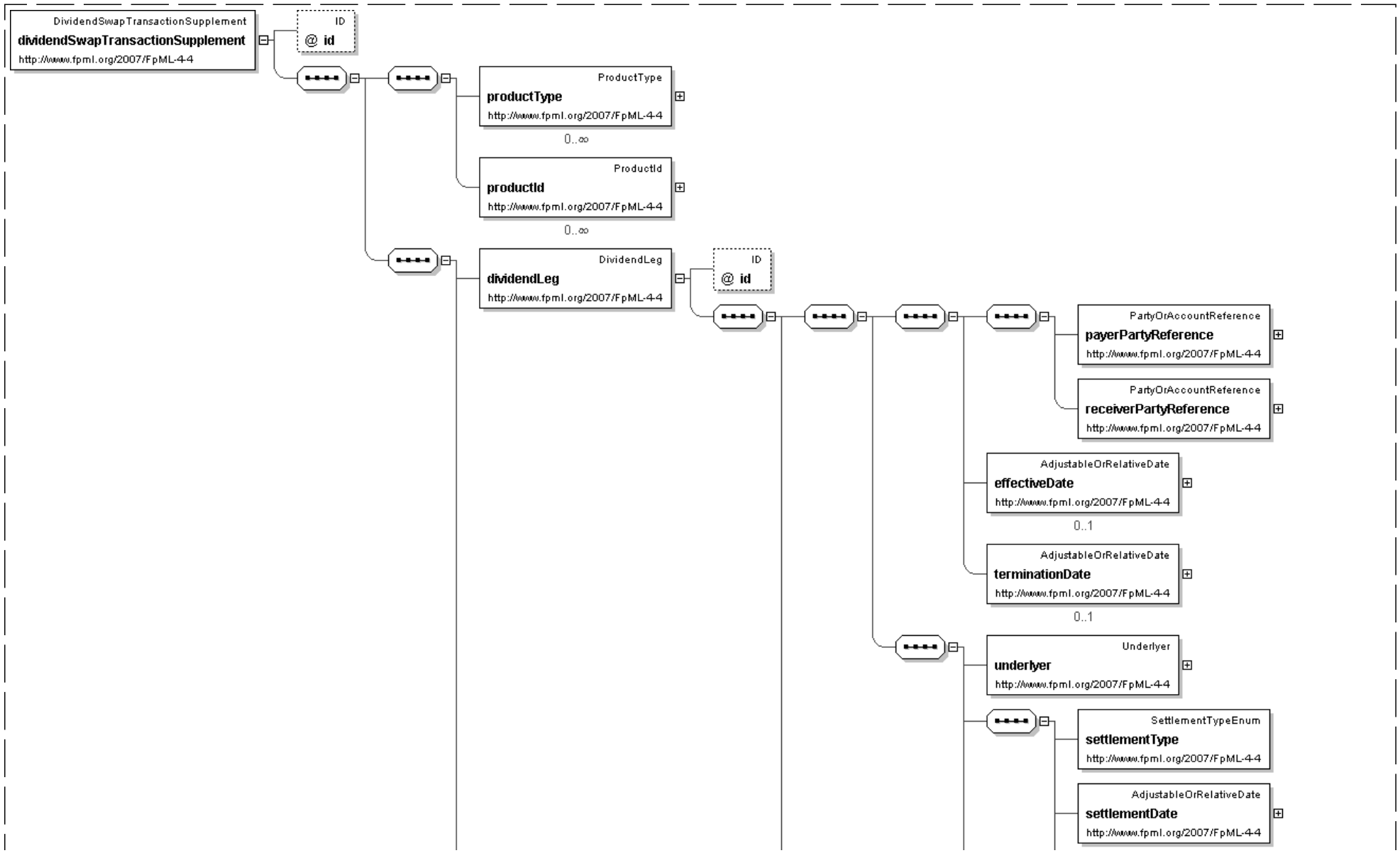
Global Declarations

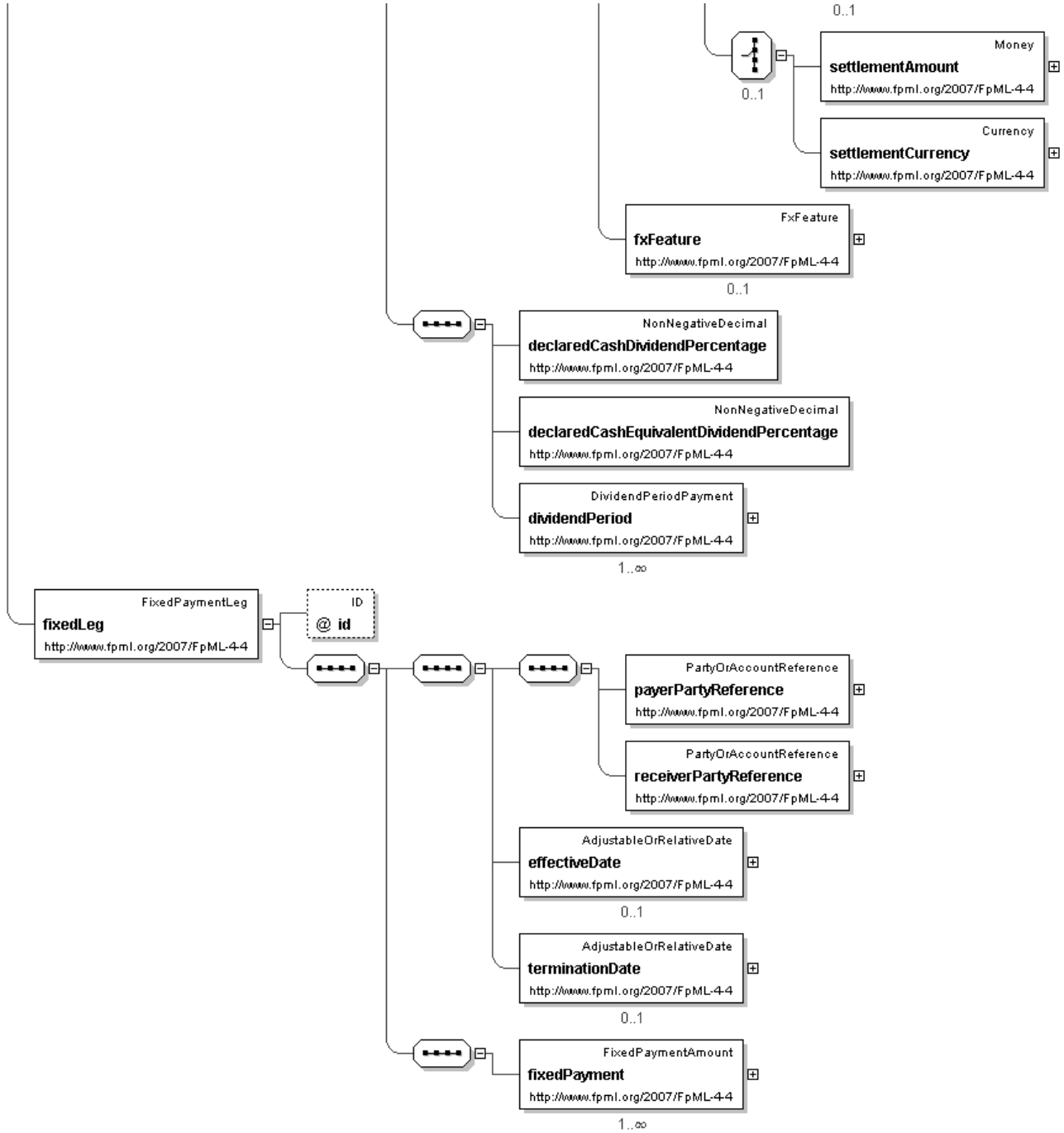
Element: dividendSwapTransactionSupplement

- *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





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]

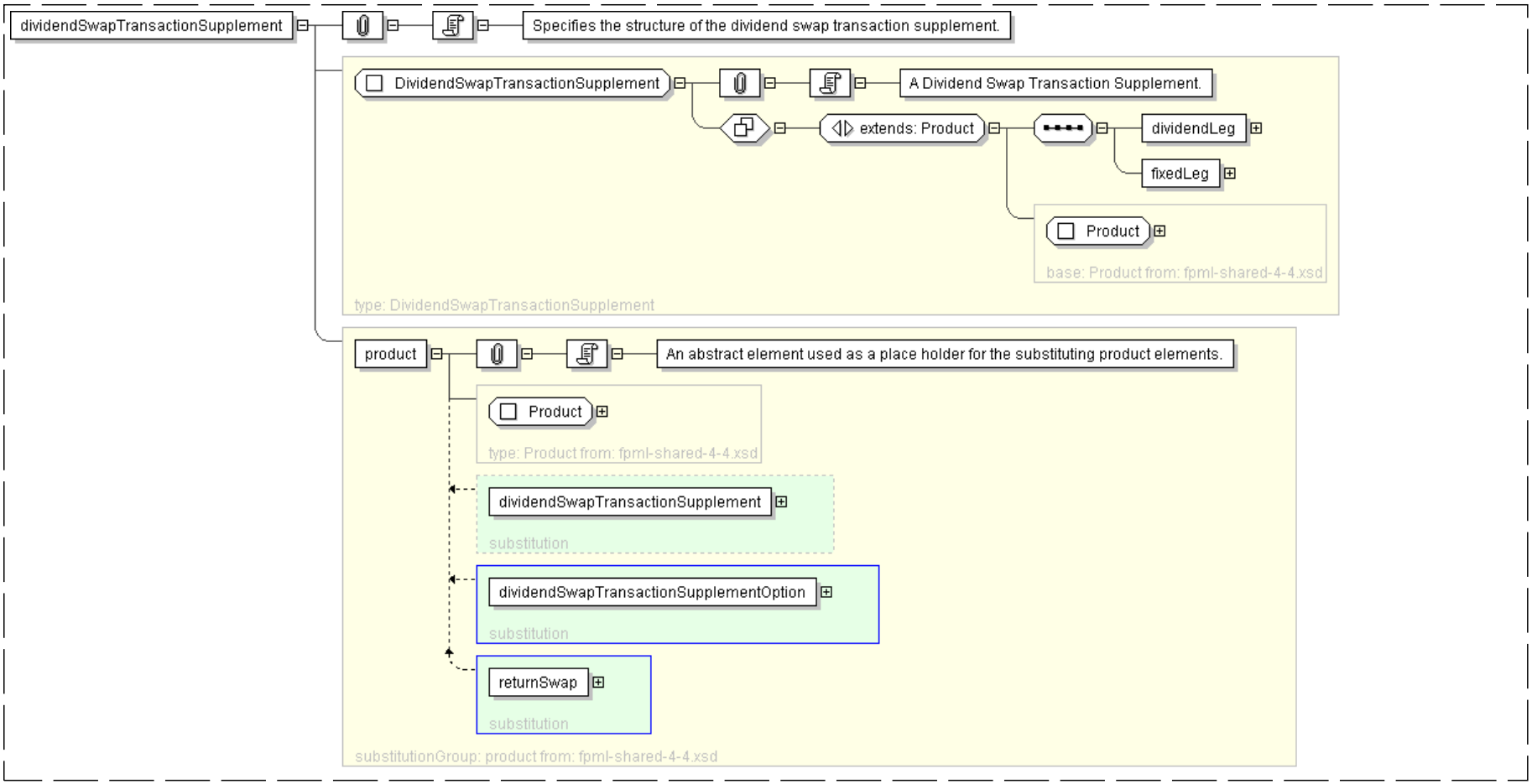
'Dividend leg.'

<fixedLeg> FixedPaymentLeg </fixedLeg> [1]

'Fixed payment leg.'

</dividendSwapTransactionSupplement>

Diagram



Schema Component Representation

```
<xsd:element name="dividendSwapTransactionSupplement" type="
DividendSwapTransactionSupplement " substitutionGroup="product"/>
```

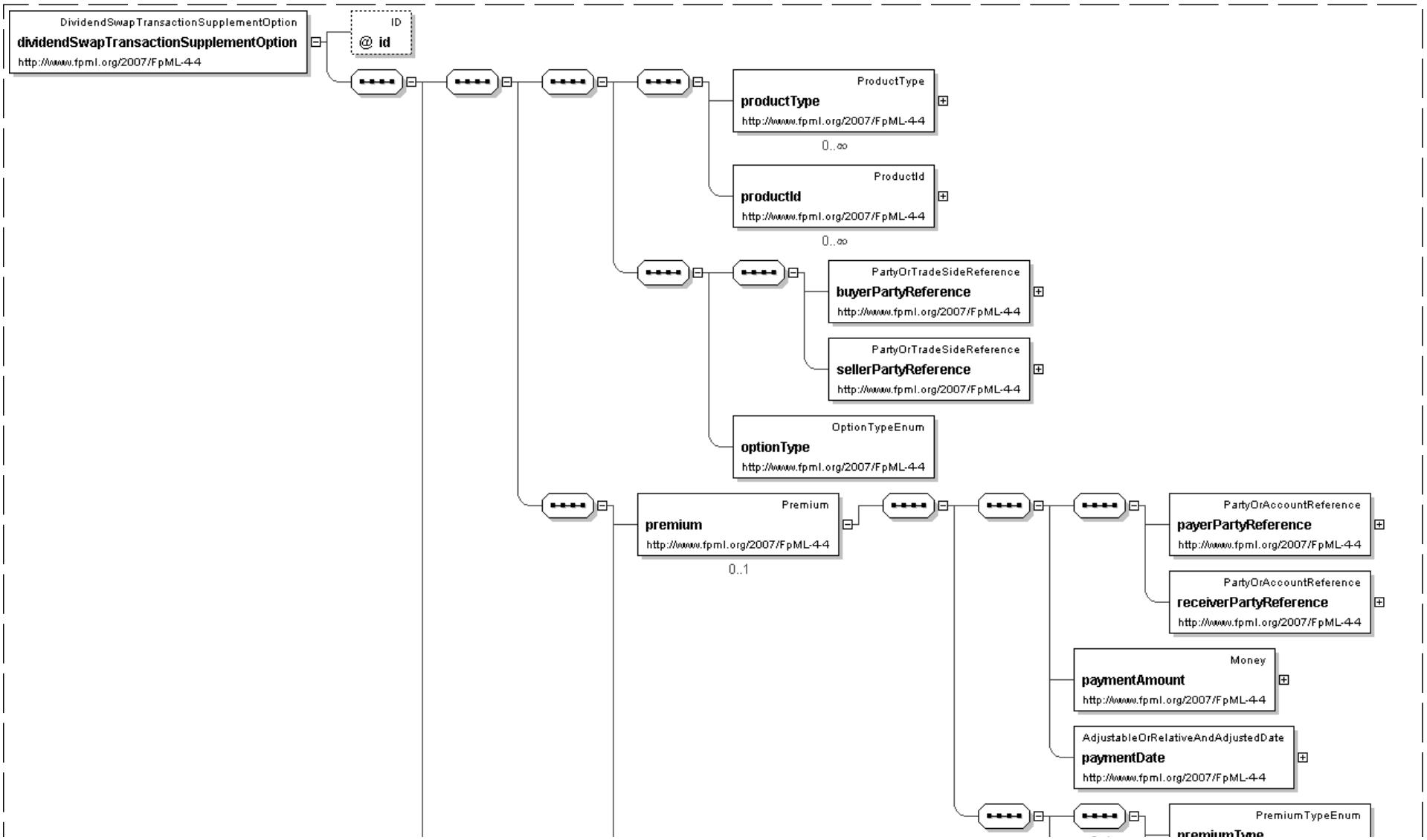
top

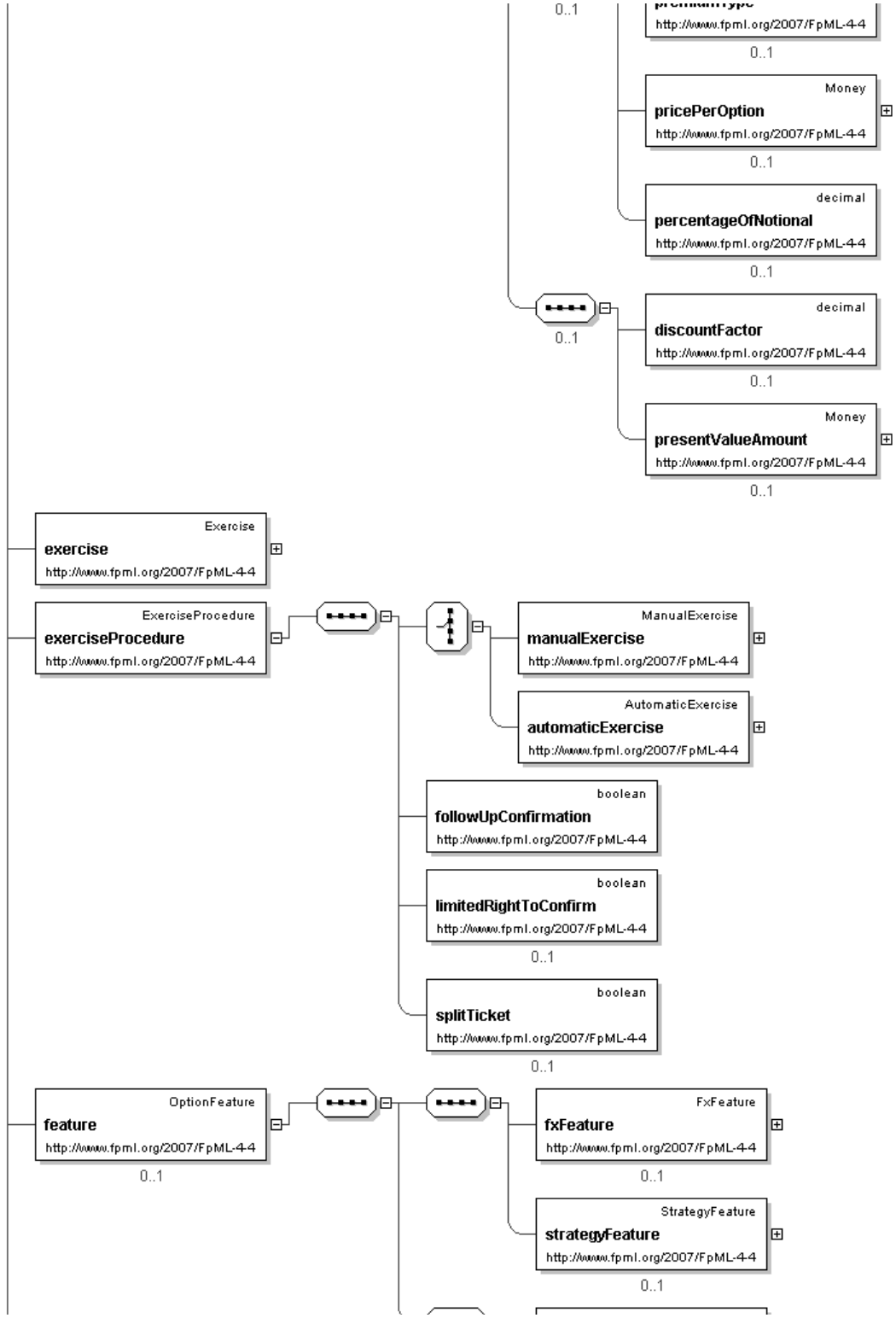
Element: dividendSwapTransactionSupplementOption

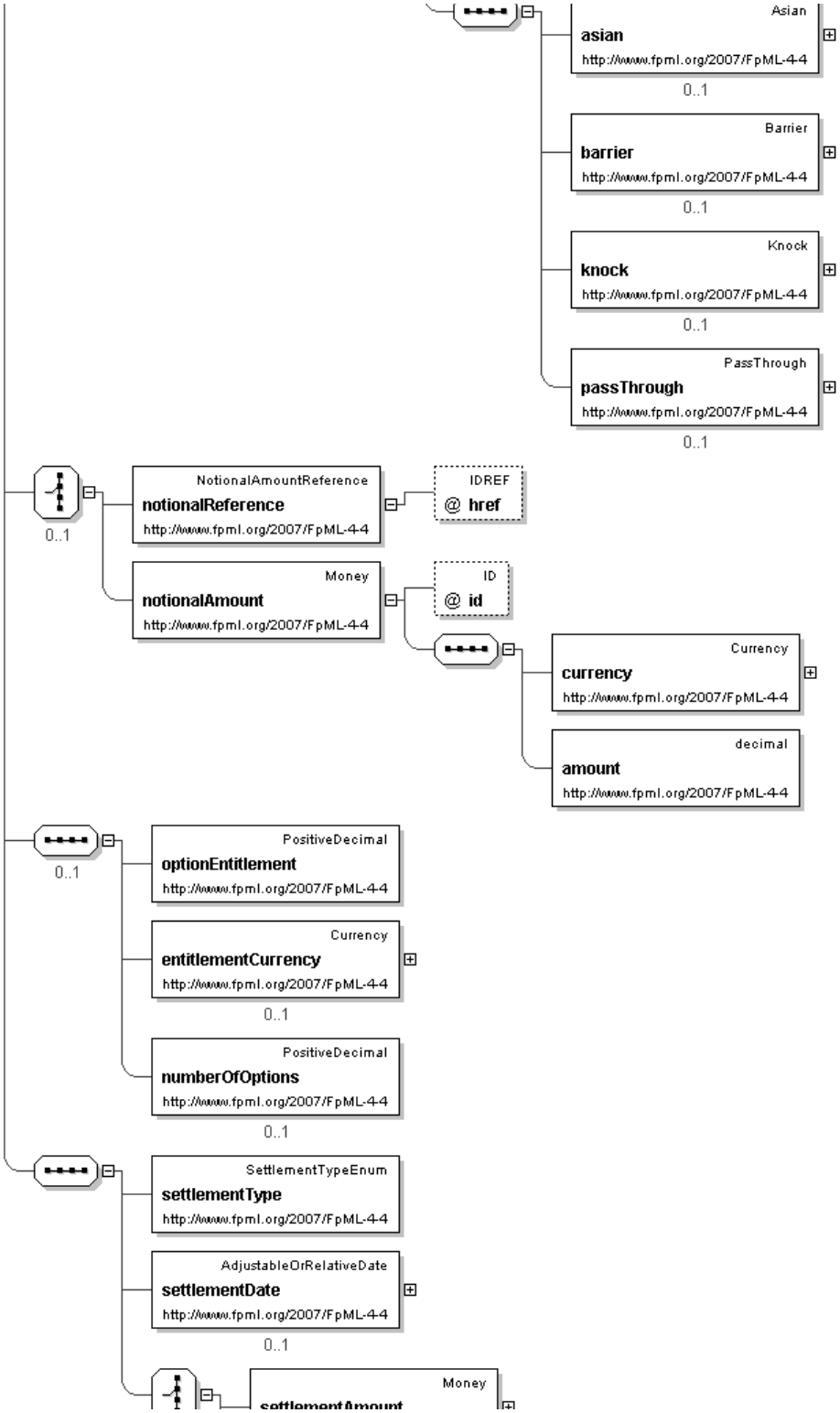
- *This element can be used wherever the following element is referenced:*
  - product

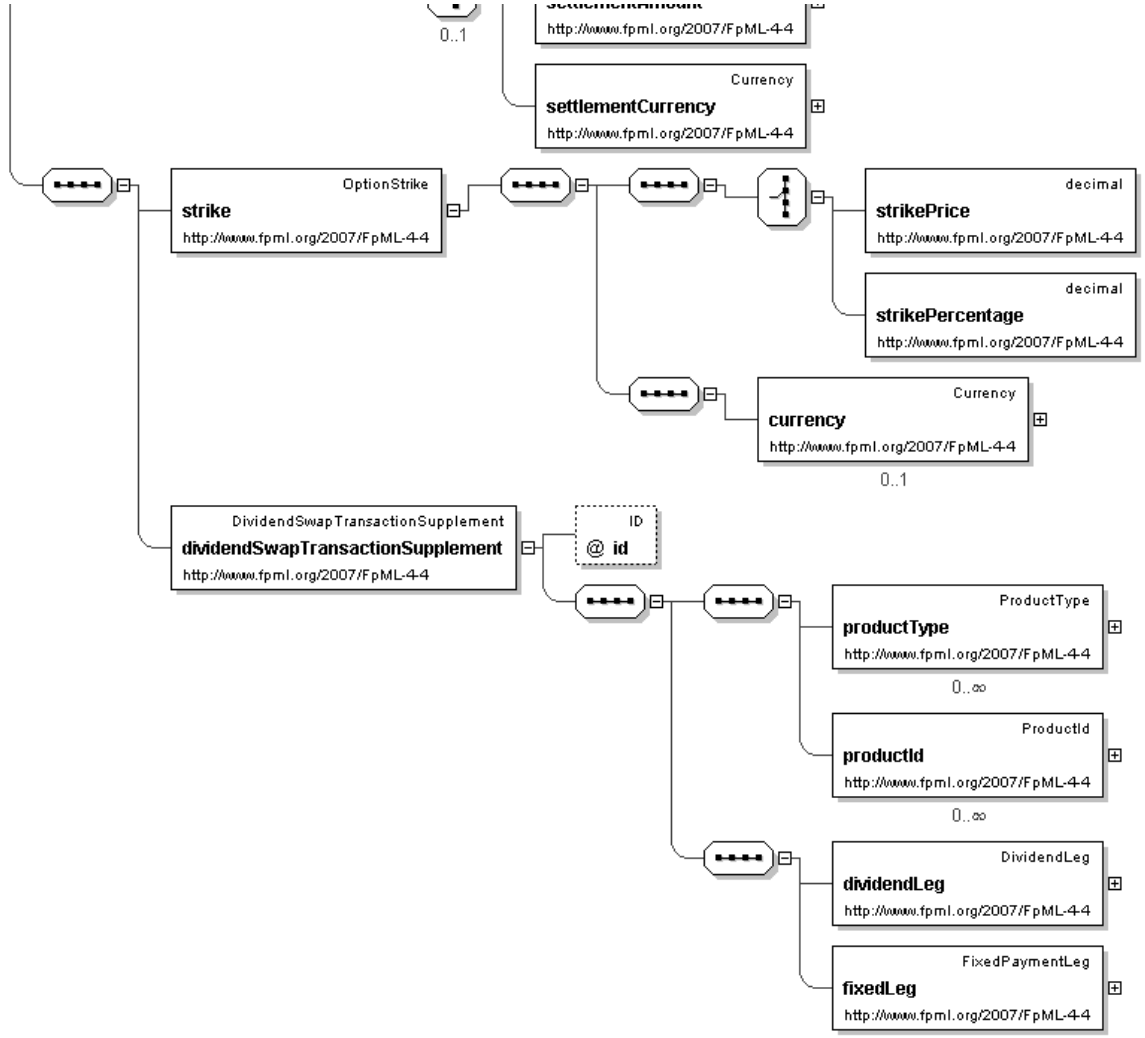
Name	dividendSwapTransactionSupplementOption
Type	DividendSwapTransactionSupplementOption
Nillable	no
Abstract	no
Documentation	Specifies the structure of the dividend swap transaction supplement option

Logical Diagram




0..1
http://www.fpml.org/2007/FpML-4-4
0..1
Money
pricePerOption
http://www.fpml.org/2007/FpML-4-4
0..1
decimal
percentageOfNotional
http://www.fpml.org/2007/FpML-4-4
0..1
0..1
decimal
discountFactor
http://www.fpml.org/2007/FpML-4-4
0..1
Money
presentValueAmount
http://www.fpml.org/2007/FpML-4-4
0..1
Exercise
exercise
http://www.fpml.org/2007/FpML-4-4
ExerciseProcedure
exerciseProcedure
http://www.fpml.org/2007/FpML-4-4
ManualExercise
manualExercise
http://www.fpml.org/2007/FpML-4-4
AutomaticExercise
automaticExercise
http://www.fpml.org/2007/FpML-4-4
boolean
followUpConfirmation
http://www.fpml.org/2007/FpML-4-4
boolean
limitedRightToConfirm
http://www.fpml.org/2007/FpML-4-4
0..1
boolean
splitTicket
http://www.fpml.org/2007/FpML-4-4
0..1
OptionFeature
feature
http://www.fpml.org/2007/FpML-4-4
0..1
FxFeature
fxFeature
http://www.fpml.org/2007/FpML-4-4
0..1
StrategyFeature
strategyFeature
http://www.fpml.org/2007/FpML-4-4
0..1


Asian
asian
http://www.fpml.org/2007/FpML-4-4
0..1
Barrier
barrier
http://www.fpml.org/2007/FpML-4-4
0..1
Knock
knock
http://www.fpml.org/2007/FpML-4-4
0..1
PassThrough
passThrough
http://www.fpml.org/2007/FpML-4-4
0..1
0..1
NotionalAmountReference
notionalReference
http://www.fpml.org/2007/FpML-4-4
IDREF
@ href
Money
notionalAmount
http://www.fpml.org/2007/FpML-4-4
ID
@ id
Currency
currency
http://www.fpml.org/2007/FpML-4-4
decimal
amount
http://www.fpml.org/2007/FpML-4-4
0..1
PositiveDecimal
optionEntitlement
http://www.fpml.org/2007/FpML-4-4
Currency
entitlementCurrency
http://www.fpml.org/2007/FpML-4-4
0..1
PositiveDecimal
numberOfOptions
http://www.fpml.org/2007/FpML-4-4
0..1
SettlementTypeEnum
settlementType
http://www.fpml.org/2007/FpML-4-4
AdjustableOrRelativeDate
settlementDate
http://www.fpml.org/2007/FpML-4-4
0..1
Money



XML Instance Representation

```
<dividendSwapTransactionSupplementOption
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> [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> OptionStrike </strike> [1]

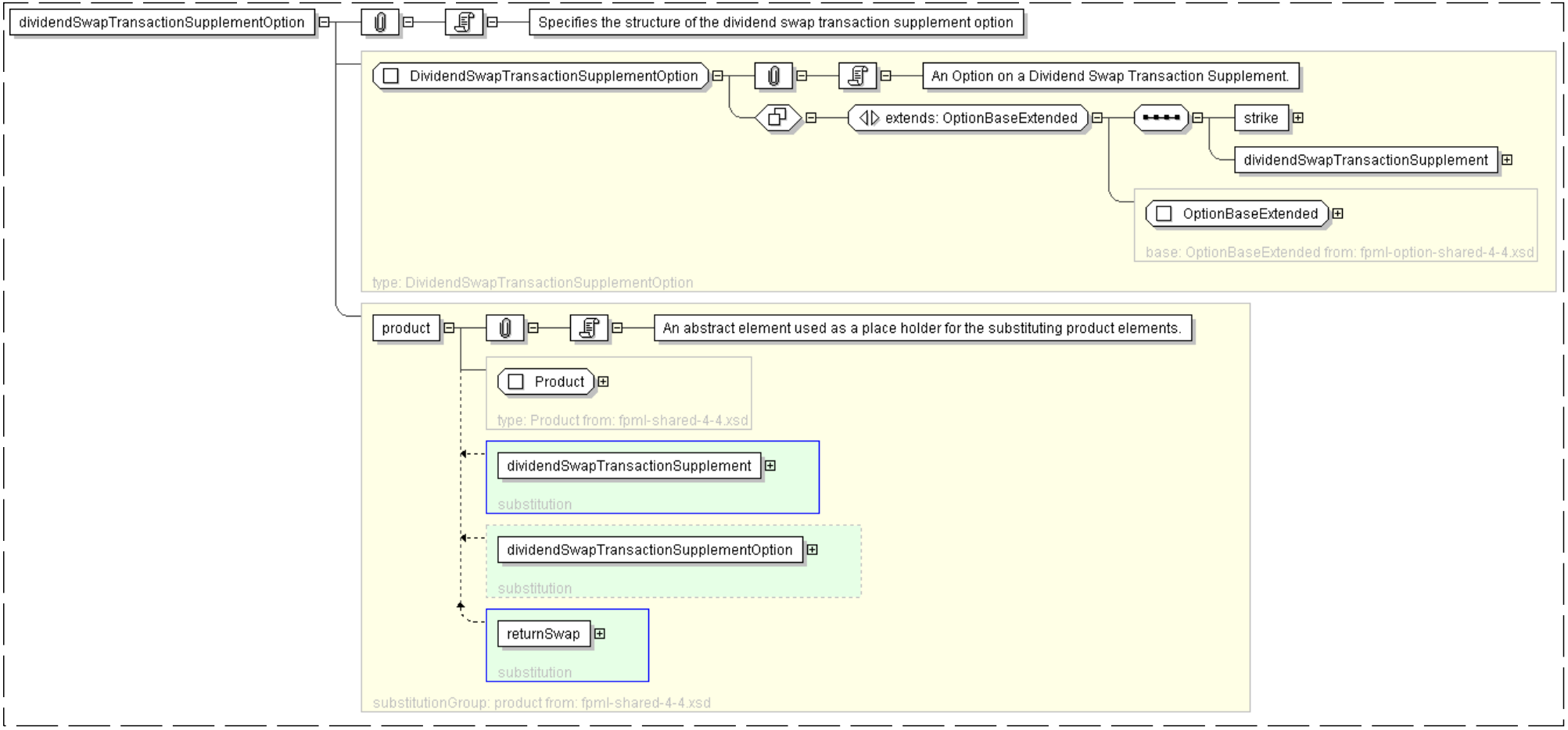
'Strike of the Dividend Swap Transaction Supplement.'

<dividendSwapTransactionSupplement> DividendSwapTransactionSupplement
</dividendSwapTransactionSupplement> [1]

'Dividend Swap Transaction Supplement which is the underlyer of this Option.'

```
</dividendSwapTransactionSupplementOption>
```

Diagram



Schema Component Representation

```
<xsd:element name="dividendSwapTransactionSupplementOption"
type=" DividendSwapTransactionSupplementOption " substitutionGroup="product"/>
```

top

Global Definitions

Complex Type: DividendLeg

Super-types:	DirectionalLegUnderlyer < **DividendLeg** (by extension)
Sub-types:	None

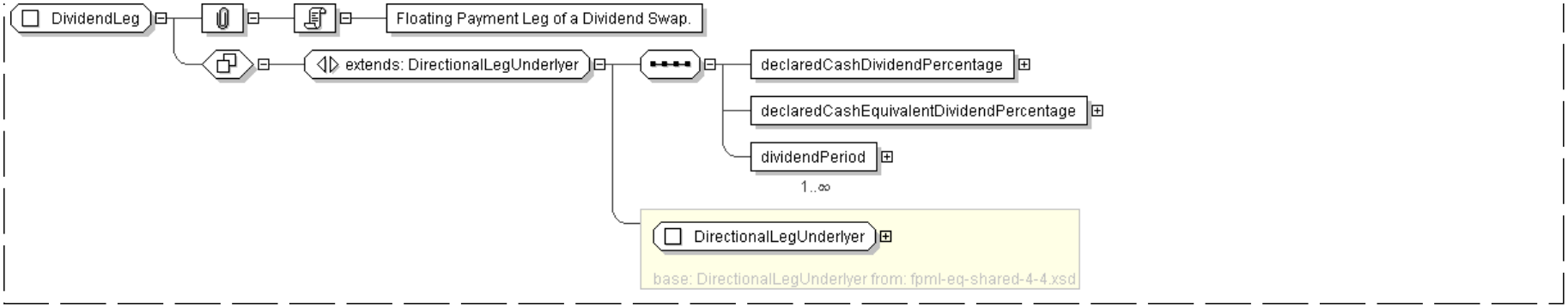
Name	DividendLeg
Used by (from the same schema document)	Complex Type **DividendSwapTransactionSupplement**
Abstract	no

Documentation Floating Payment Leg of a Dividend Swap.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <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> [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> [1]
   'Declared Cash Dividend Percentage.'

   <declaredCashEquivalentDividendPercentage> NonNegativeDecimal
   </declaredCashEquivalentDividendPercentage> [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.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendLeg">
   <xsd:complexContent>
      <xsd:extension base=" DirectionalLegUnderlyer ">
         <xsd:sequence>
            <xsd:element name="declaredCashDividendPercentage" type=" NonNegativeDecimal "/>
            <xsd:element name="declaredCashEquivalentDividendPercentage" type=" NonNegativeDecimal "/>
            <xsd:element name="dividendPeriod" type=" DividendPeriodPayment " maxOccurs="unbounded"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DividendPeriodPayment

Super-types:	DividendPeriod < **DividendPeriodPayment** (by extension)
Sub-types:	None

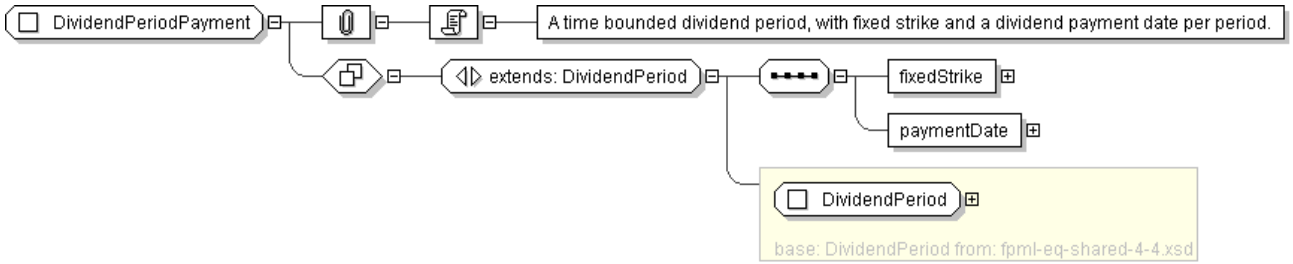
Name	DividendPeriodPayment
Used by (from the same schema document)	Complex Type **DividendLeg**
Abstract	no
Documentation	A time bounded dividend period, with fixed strike and a dividend payment date per period.

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.'

</...>
```

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

top

Complex Type: DividendSwapTransactionSupplement

Super-types:	Product < **DividendSwapTransactionSupplement** (by extension)
Sub-types:	None

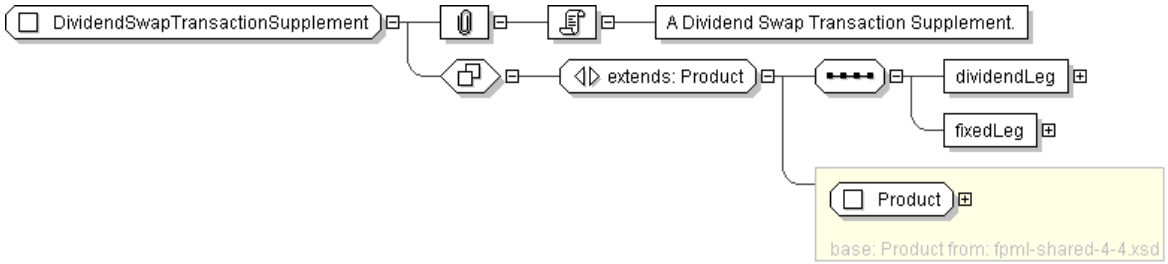
Name	DividendSwapTransactionSupplement
Used by (from the same schema document)	Complex Type **DividendSwapTransactionSupplementOption** , Element **dividendSwapTransactionSupplement**
Abstract	no
Documentation	A Dividend Swap Transaction Supplement.

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.'

</...>
```

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

top

Complex Type: DividendSwapTransactionSupplementOption

Super-types:	OptionBaseExtended < **DividendSwapTransactionSupplementOption** (by extension)
Sub-types:	None

Name	DividendSwapTransactionSupplementOption
Used by (from the same schema document)	Element **dividendSwapTransactionSupplementOption**
Abstract	no
Documentation	An Option on 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.'
```

<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> [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

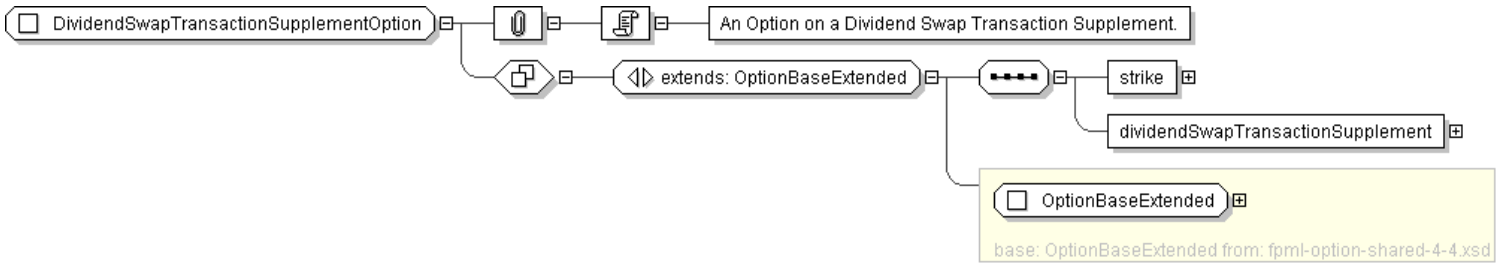
End Group: SettlementAmountOrCurrency.model

<strike> OptionStrike </strike> [1]

'Strike of the Dividend Swap Transaction Supplement.'

```
    <dividendSwapTransactionSupplement> DividendSwapTransactionSupplement
    </dividendSwapTransactionSupplement> [1]
    'Dividend Swap Transaction Supplement which is the underlyer of this Option.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendSwapTransactionSupplementOption">
   <xsd:complexContent>
      <xsd:extension base=" OptionBaseExtended ">
         <xsd:sequence>
            <xsd:element name="strike" type=" OptionStrike "/>
            <xsd:element name="dividendSwapTransactionSupplement" type="
            DividendSwapTransactionSupplement "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: FixedPaymentAmount

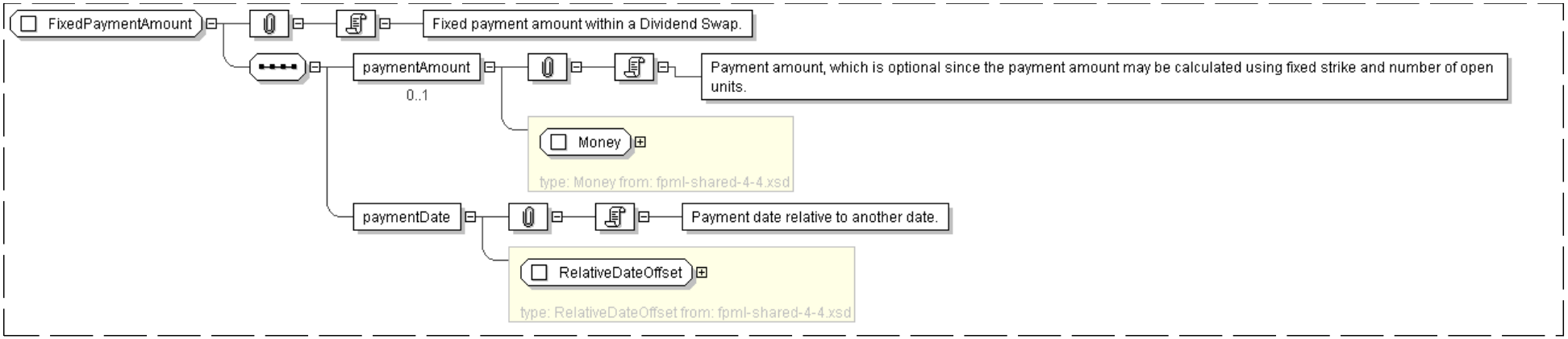
Super-types:	None
Sub-types:	None

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

```
<...>
    <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:sequence>
      <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
      <xsd:element name="paymentDate" type=" RelativeDateOffset "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: FixedPaymentLeg

Super-types:	DirectionalLeg < **FixedPaymentLeg** (by extension)
Sub-types:	None

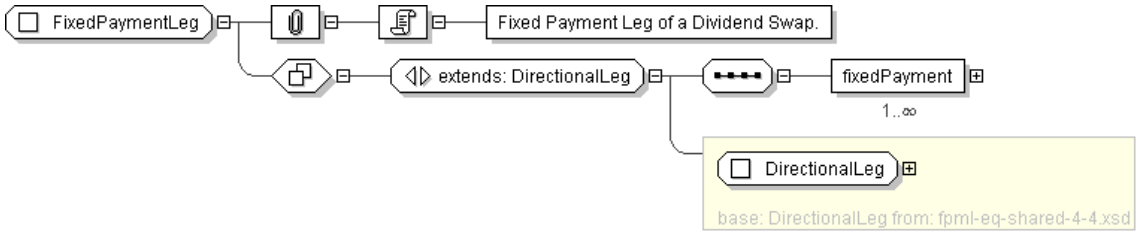
Name	FixedPaymentLeg
Used by (from the same schema document)	Complex Type **DividendSwapTransactionSupplement**
Abstract	no
Documentation	Fixed Payment Leg of a Dividend Swap.

XML Instance Representation

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

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.