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: **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/2007/FpML-4-4**

Version	$Revision: 2350 $
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-4-4.xsd ○ fpml-asset-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
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: 2350 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-doc-4-4.xsd"/>
   <xsd:include schemaLocation="fpml-asset-4-4.xsd"/>
   ...
</xsd:schema>
```

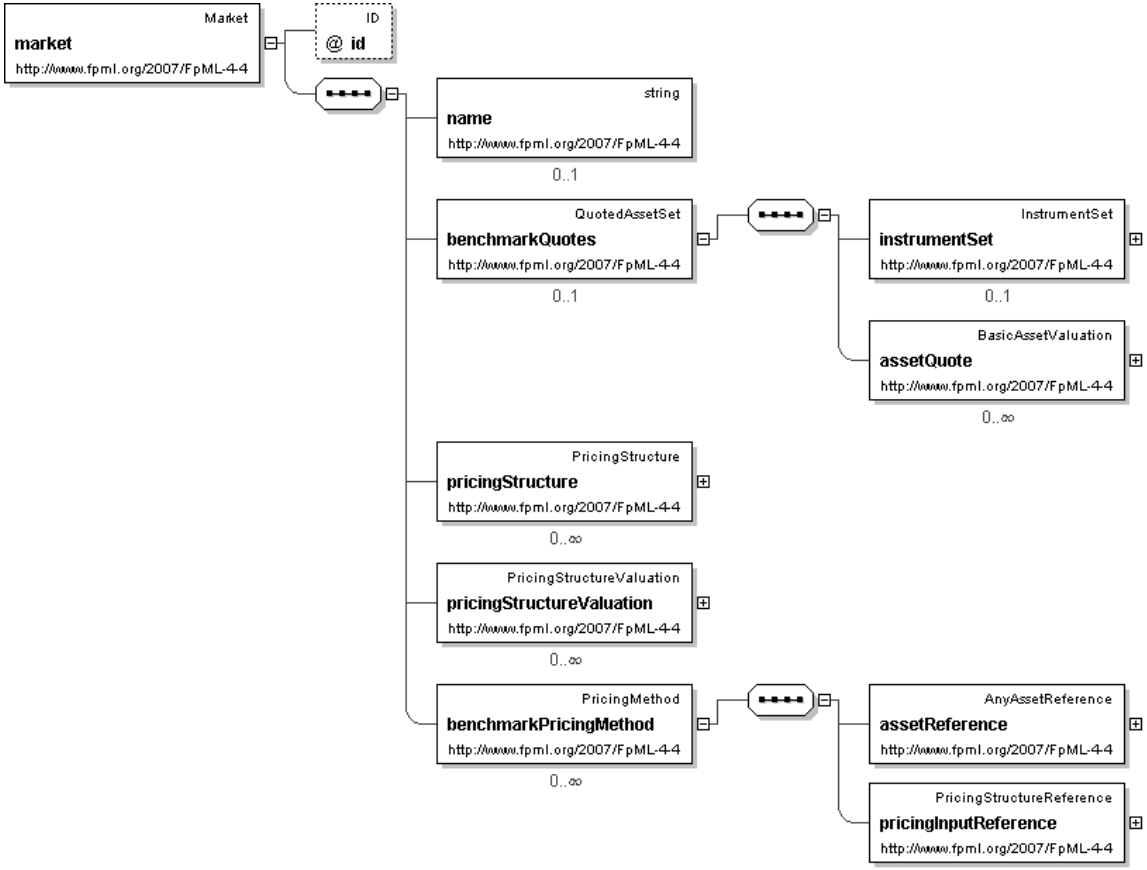
top

Global Declarations

Element: market

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.

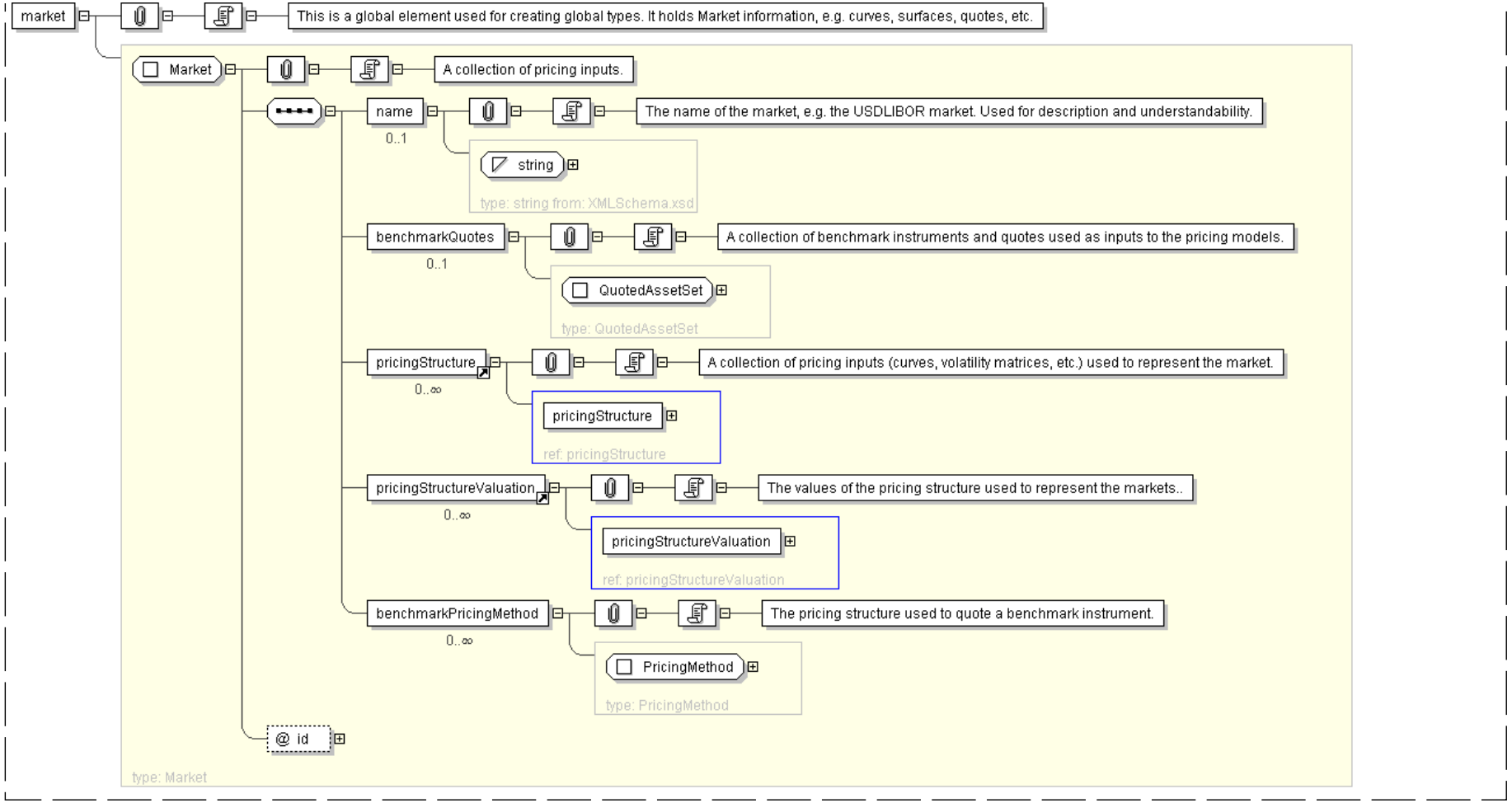
Logical Diagram



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

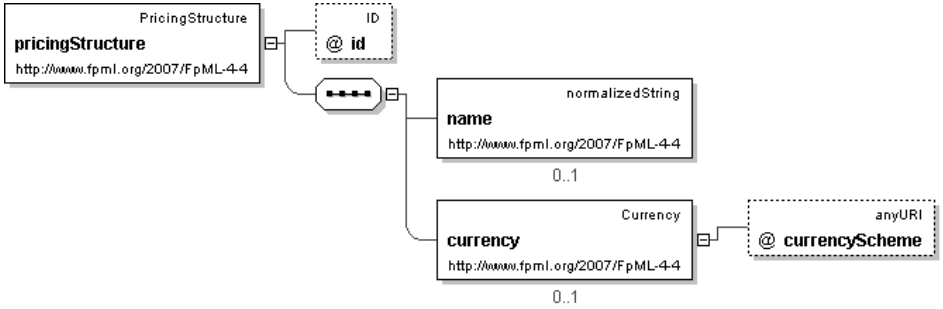
```
<xsd:element name="market" type=" Market "/>
```

top

Element: pricingStructure

Name	pricingStructure
Used by (from the same schema document)	Complex Type **Market**
Type	PricingStructure
Nillable	no
Abstract	yes

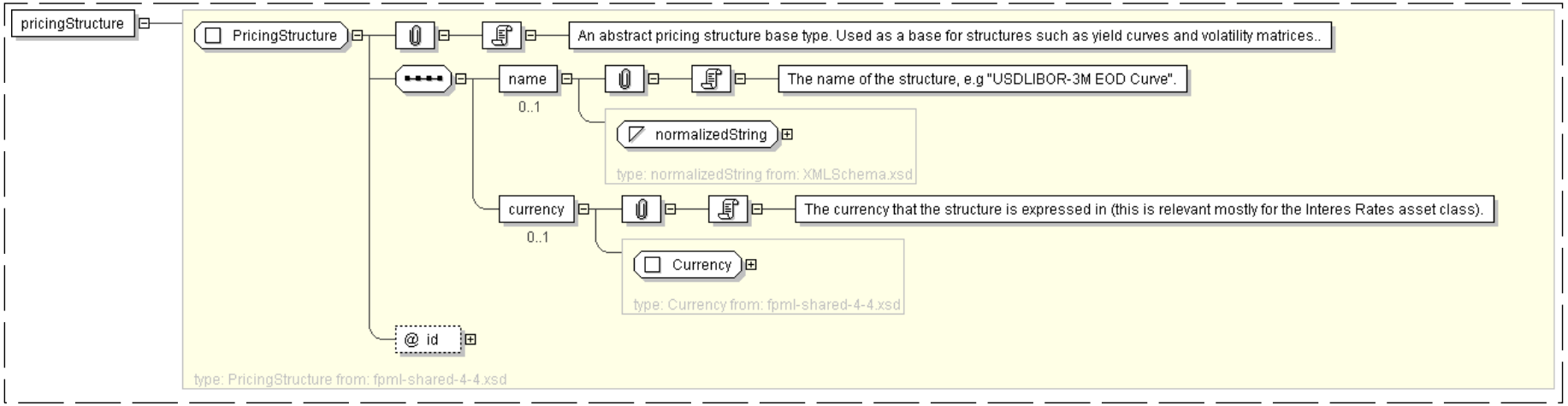
Logical Diagram



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

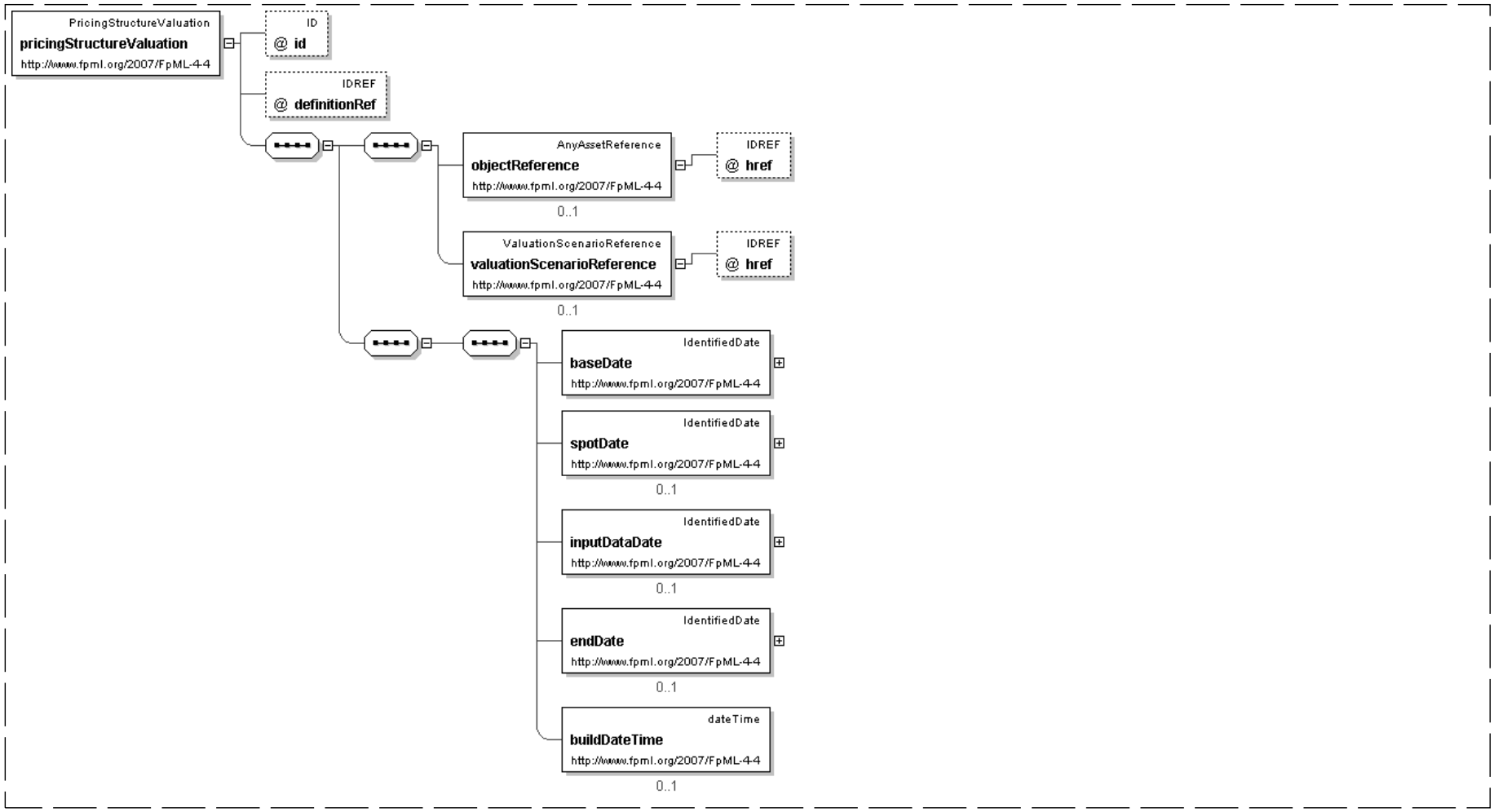
```
<xsd:element name="pricingStructure" type=" PricingStructure " abstract="true"/>
```

top

Element: pricingStructureValuation

Name	pricingStructureValuation
Used by (from the same schema document)	Complex Type **Market**
Type	PricingStructureValuation
Nillable	no
Abstract	yes

Logical Diagram

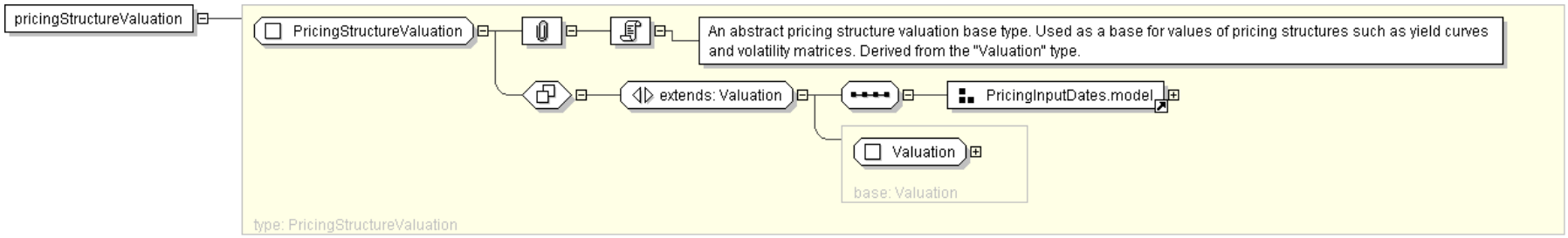


XML Instance Representation

```
<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"/>
```

top

Global Definitions

Complex Type: AssetOrTermPointOrPricingStructureReference

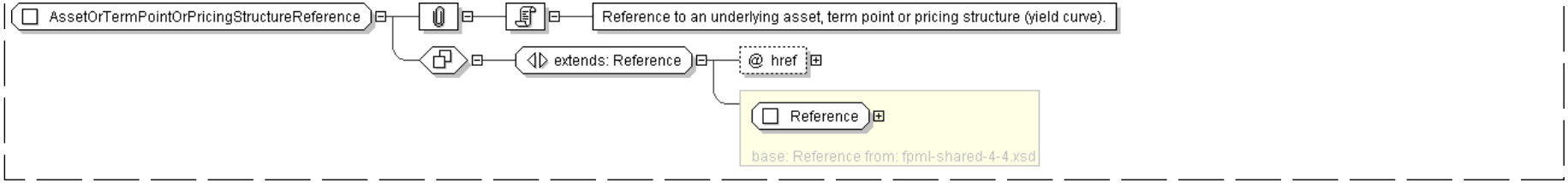
Super-types:	Reference < **AssetOrTermPointOrPricingStructureReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: BasicAssetValuation

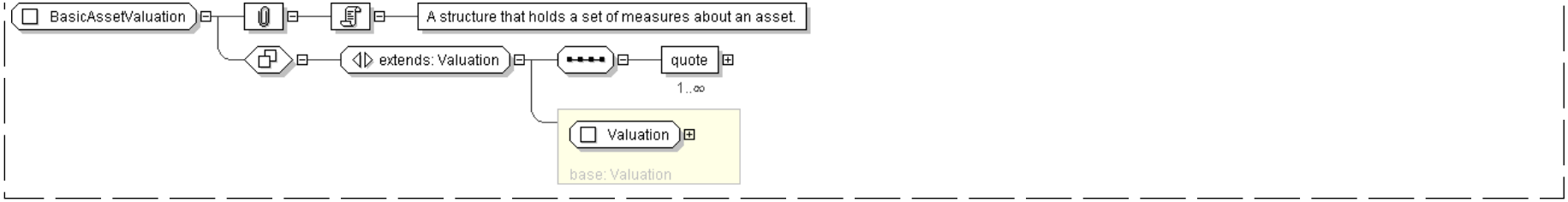
Super-types:	Valuation < **BasicAssetValuation** (by extension)
Sub-types:	None

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>
```

top

Complex Type: DenominatorTerm

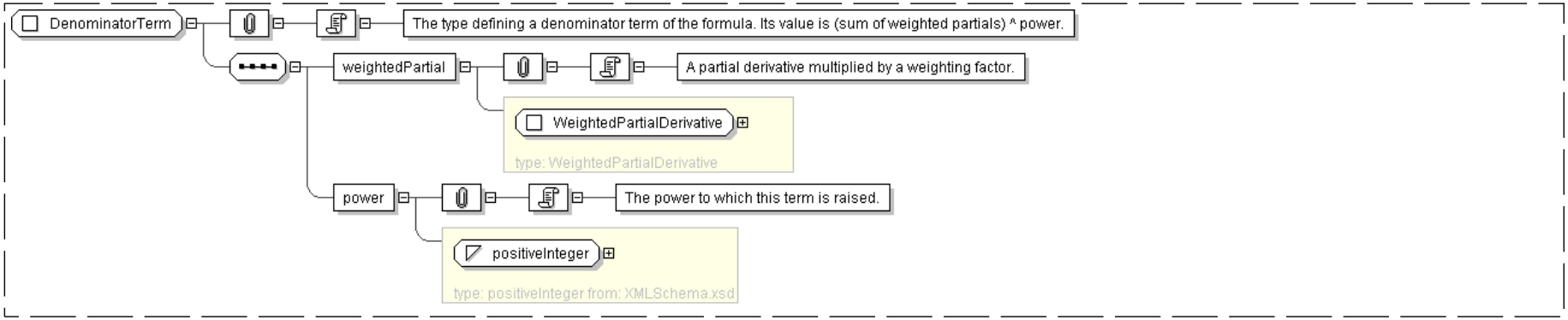
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: DerivativeCalculationMethod

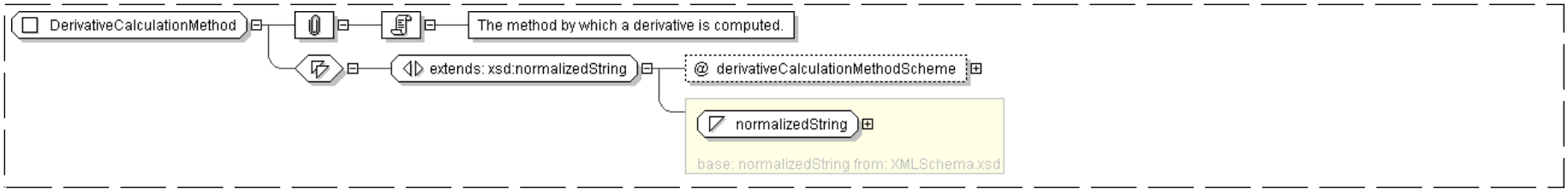
Super-types:	xsd:normalizedString < **DerivativeCalculationMethod** (by extension)
Sub-types:	None

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]">
xsd:normalizedString
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DerivativeCalculationMethod">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:normalizedString ">
         <xsd:attribute name="derivativeCalculationMethodScheme" type=" xsd:anyURI " default="http://
         www.fpml.org/coding-scheme/derivative-calculation-method-1-0"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

top

Complex Type: DerivativeCalculationProcedure

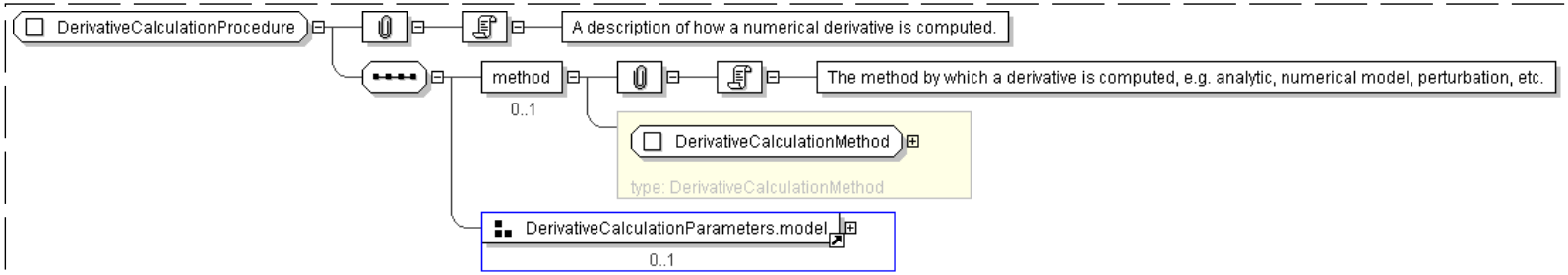
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: DerivativeFormula

Super-types:	None
Sub-types:	None

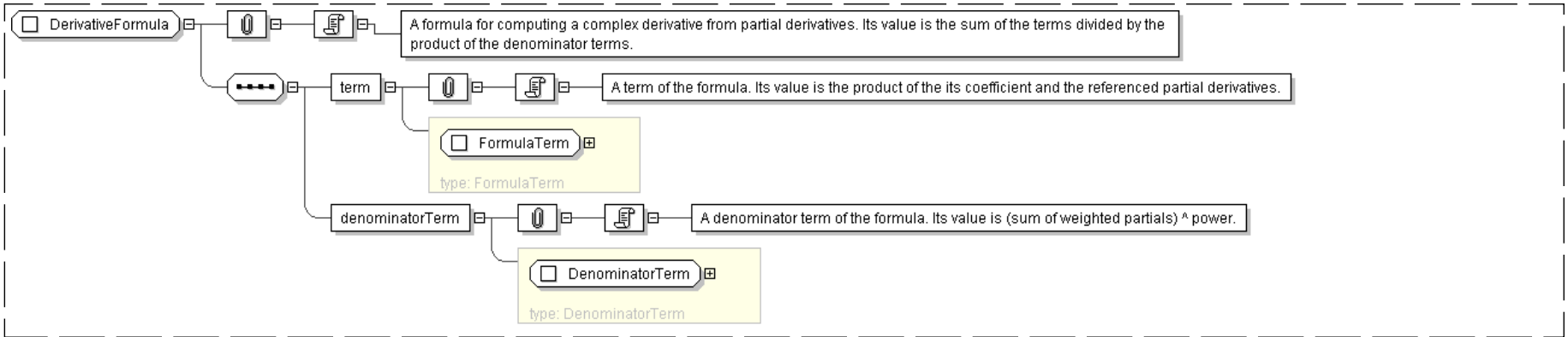
Name	DerivativeFormula

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>
```

top

Complex Type: FormulaTerm

Super-types:	None
Sub-types:	None

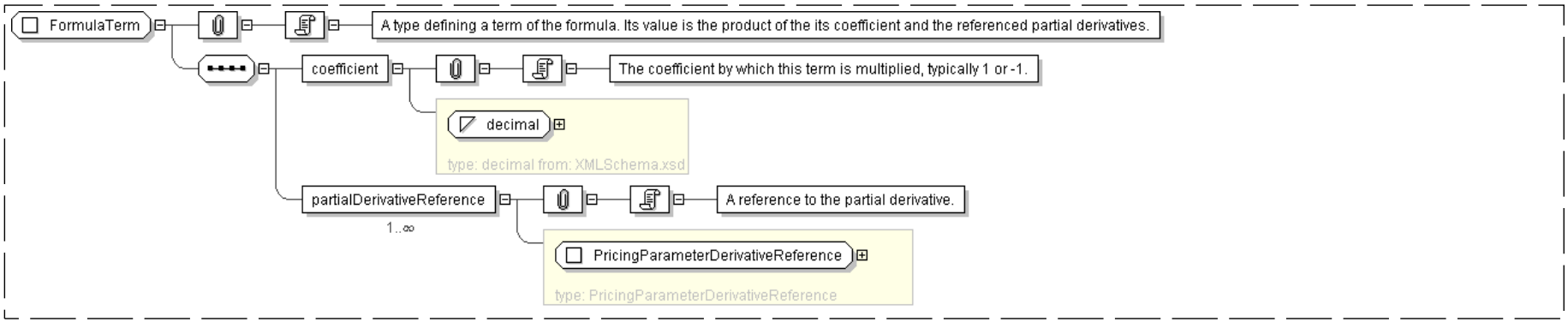
Name	FormulaTerm
Used by (from the same schema document)	Complex Type **DerivativeFormula**
Abstract	no
Documentation	A type defining a term of the formula. Its value is the product of the its coefficient and the referenced partial derivatives.

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>
```

top

Complex Type: GenericDimension

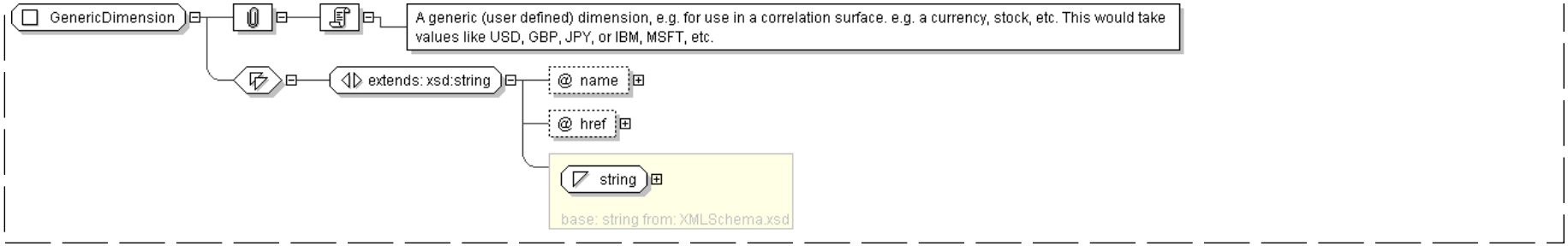
Super-types:	xsd:string < **GenericDimension** (by extension)
Sub-types:	None

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>
```

top

Complex Type: InstrumentSet

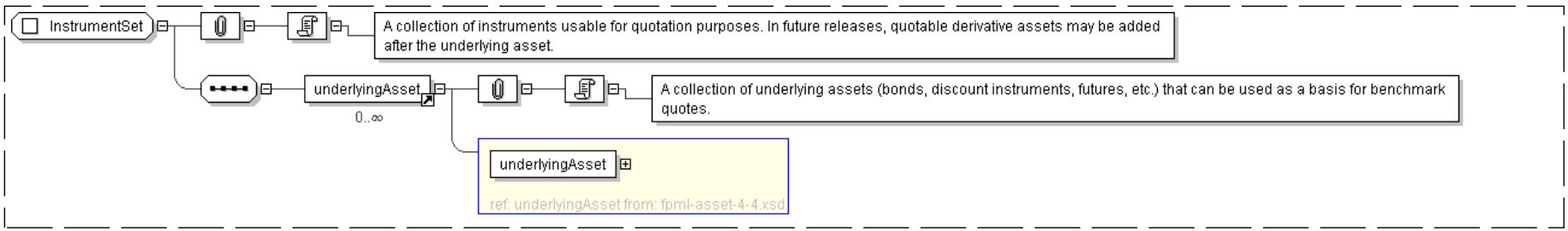
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: Market

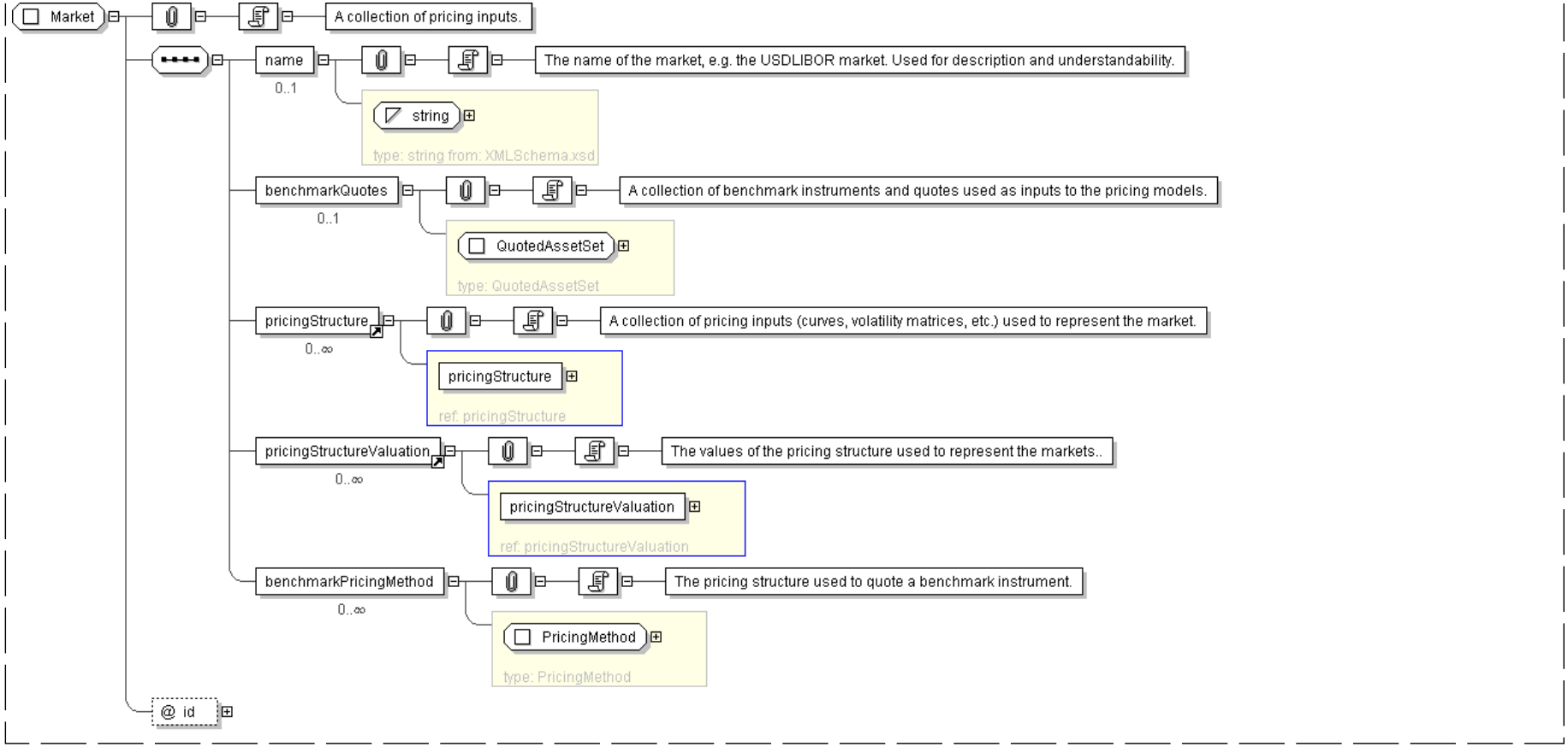
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: MarketReference

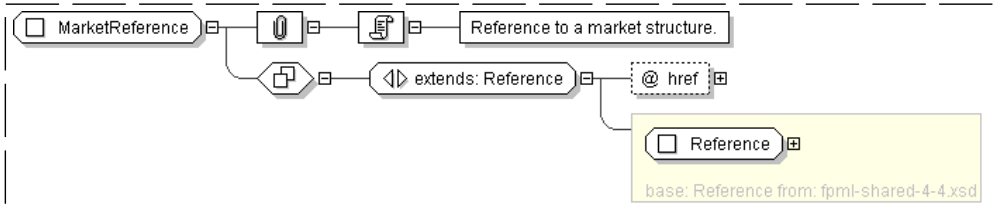
Super-types:	Reference < **MarketReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: PerturbationType

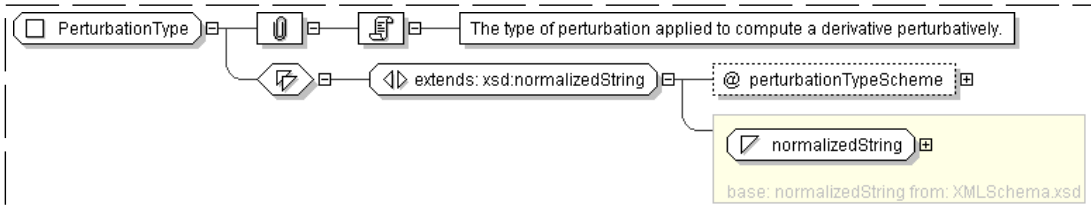
Super-types:	xsd:normalizedString < **PerturbationType** (by extension)
Sub-types:	None

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]">
xsd:normalizedString
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PerturbationType">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:normalizedString ">
         <xsd:attribute name="perturbationTypeScheme" type=" xsd:anyURI " default="http://www.fpml.
         org/coding-scheme/perturbation-type-1-0"/>
      </xsd:extension>
```

```
   </xsd:simpleContent>
</xsd:complexType>
```

top

Complex Type: PositionId

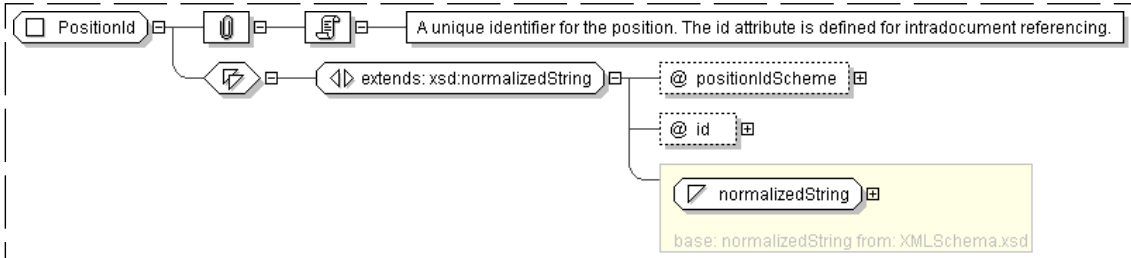
Super-types:	xsd:normalizedString < **PositionId** (by extension)
Sub-types:	None

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]">
xsd:normalizedString
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PositionId">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:normalizedString ">
         <xsd:attribute name="positionIdScheme" type=" xsd:anyURI "/>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

top

Complex Type: PricingDataPointCoordinate

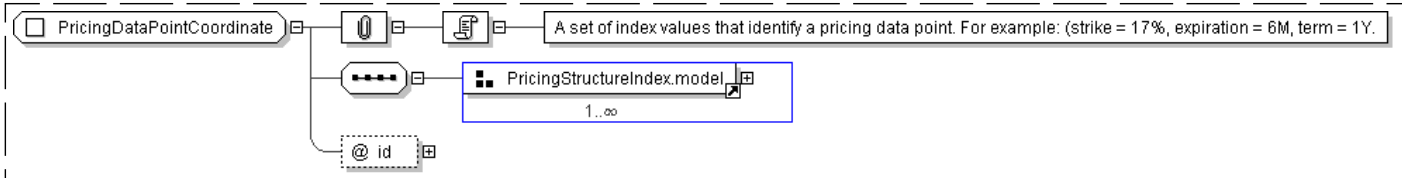
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: PricingDataPointCoordinateReference

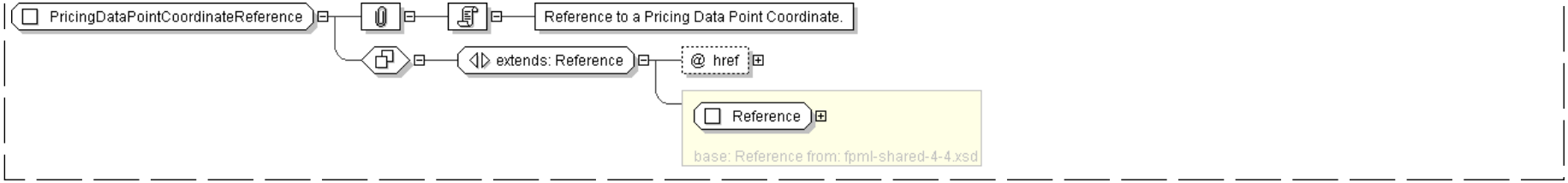
Super-types:	Reference < **PricingDataPointCoordinateReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: PricingInputReplacement

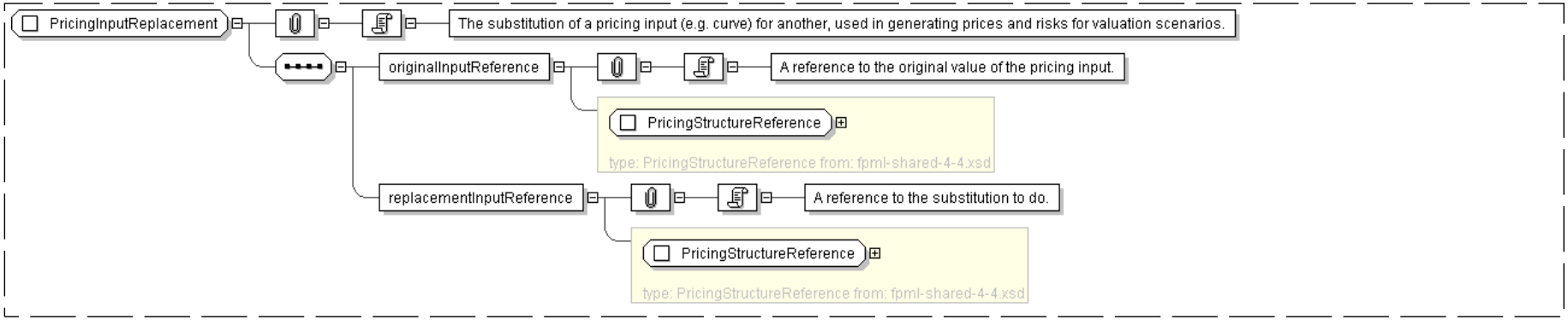
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: PricingInputType

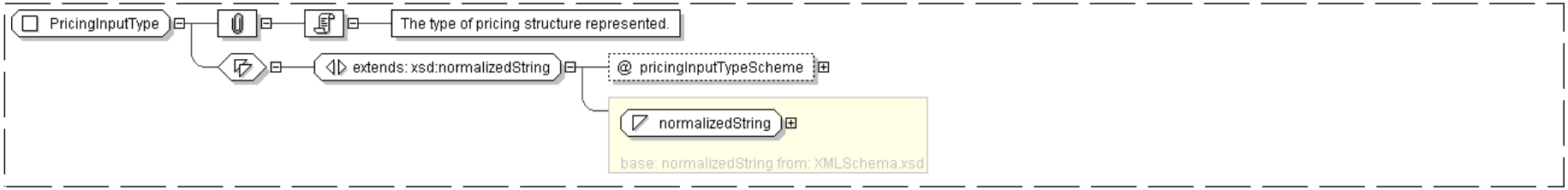
Super-types:	xsd:normalizedString < **PricingInputType** (by extension)
Sub-types:	None

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]">
xsd:normalizedString
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PricingInputType">
   <xsd:simpleContent>
      <xsd:extension base=" xsd:normalizedString ">
         <xsd:attribute name="pricingInputTypeScheme" type=" xsd:anyURI " default="http://www.fpml.
         org/coding-scheme/pricing-input-type-1-0"/>
      </xsd:extension>
   </xsd:simpleContent>
</xsd:complexType>
```

top

Complex Type: PricingMethod

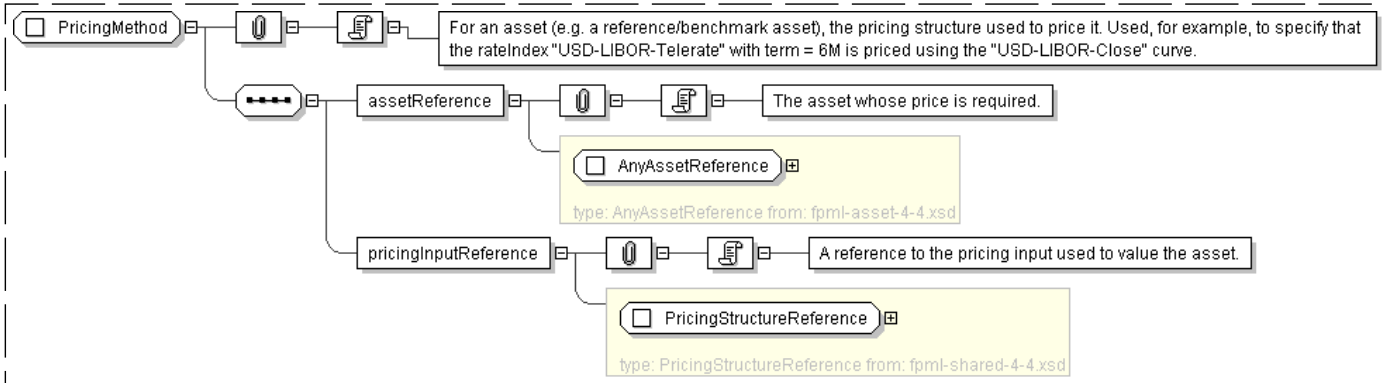
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: PricingParameterDerivative

Super-types:	None
Sub-types:	None

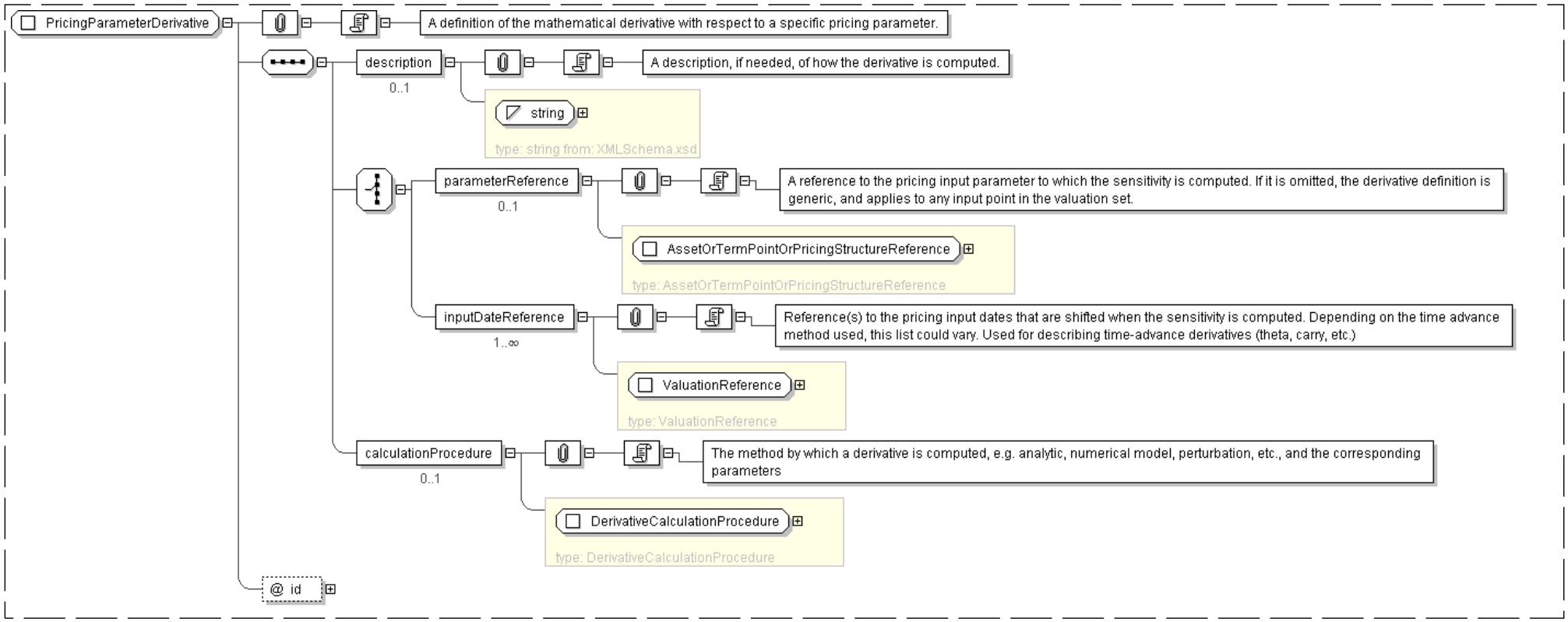
Name	PricingParameterDerivative
Used by (from the same schema document)	Model Group **ComputedDerivative.model**
Abstract	no
Documentation	A definition of the mathematical derivative with respect to a specific pricing parameter.

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>
```

top

Complex Type: PricingParameterDerivativeReference

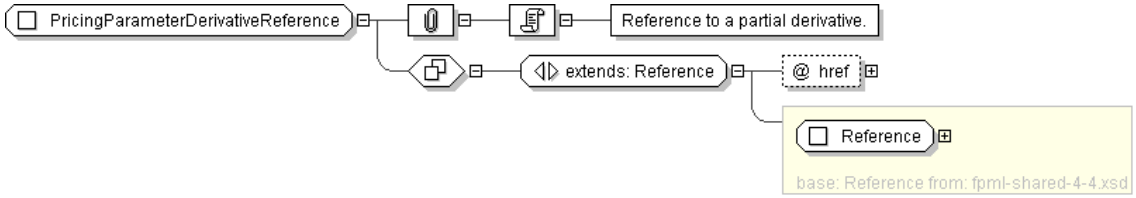
Super-types:	Reference < **PricingParameterDerivativeReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: PricingParameterShift

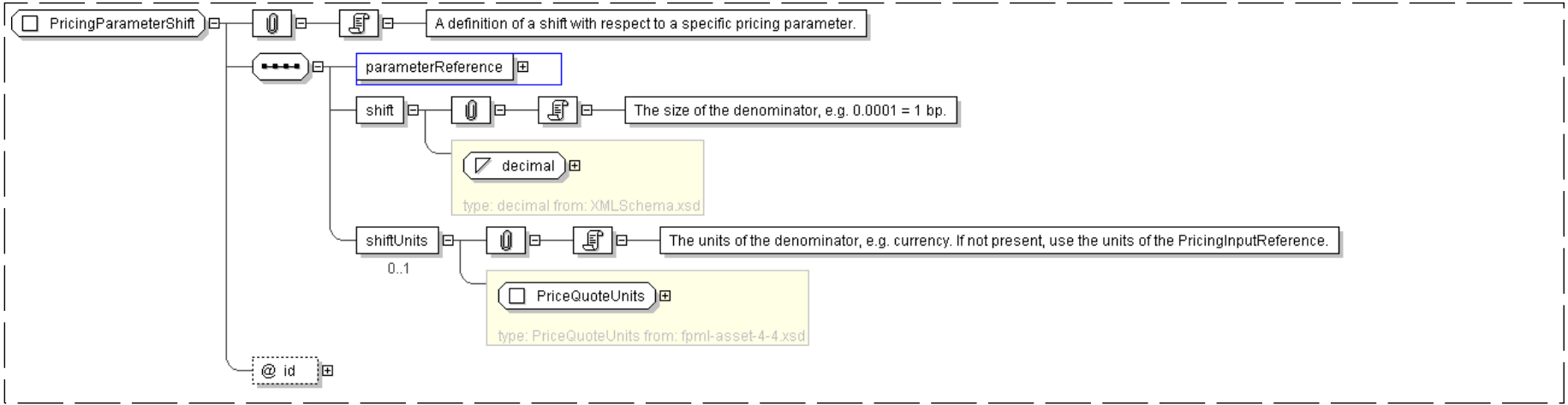
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: PricingStructureValuation

Super-types:	Valuation < **PricingStructureValuation** (by extension)
Sub-types:	None

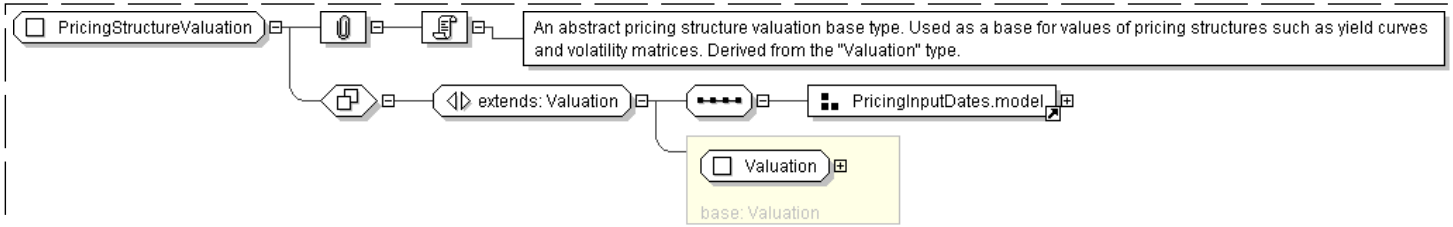
Name	PricingStructureValuation
Used by (from the same schema document)	Element **pricingStructureValuation**
Abstract	no
Documentation	An abstract pricing structure valuation base type. Used as a base for values of pricing structures such as yield curves and volatility matrices. Derived from the "Valuation" 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>
```

top

Complex Type: QuotedAssetSet

Super-types:	None
Sub-types:	None

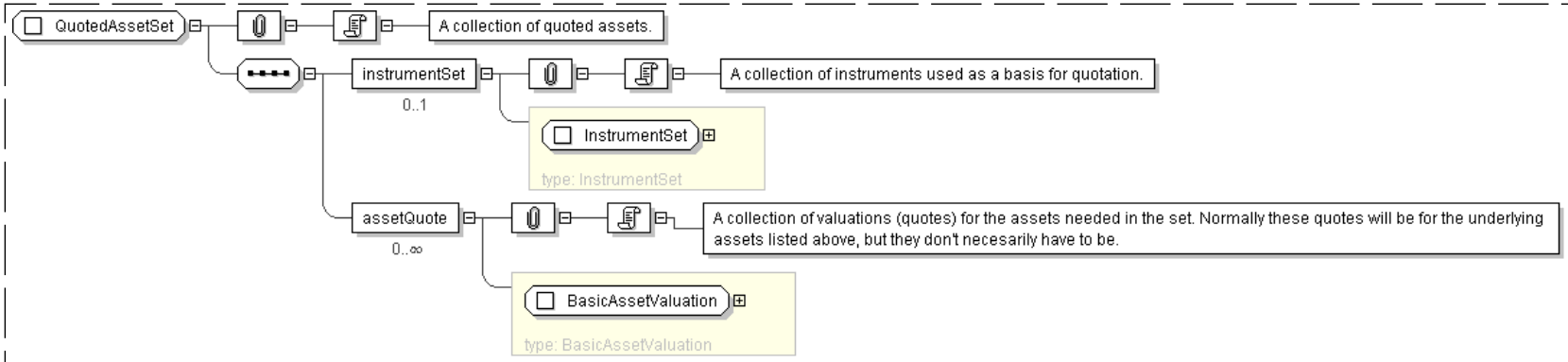
Name	QuotedAssetSet
Used by (from the same schema document)	Complex Type **Market**
Abstract	no
Documentation	A collection of quoted assets.

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>
```

top

Complex Type: SensitivityDefinition

Super-types:	None
Sub-types:	None

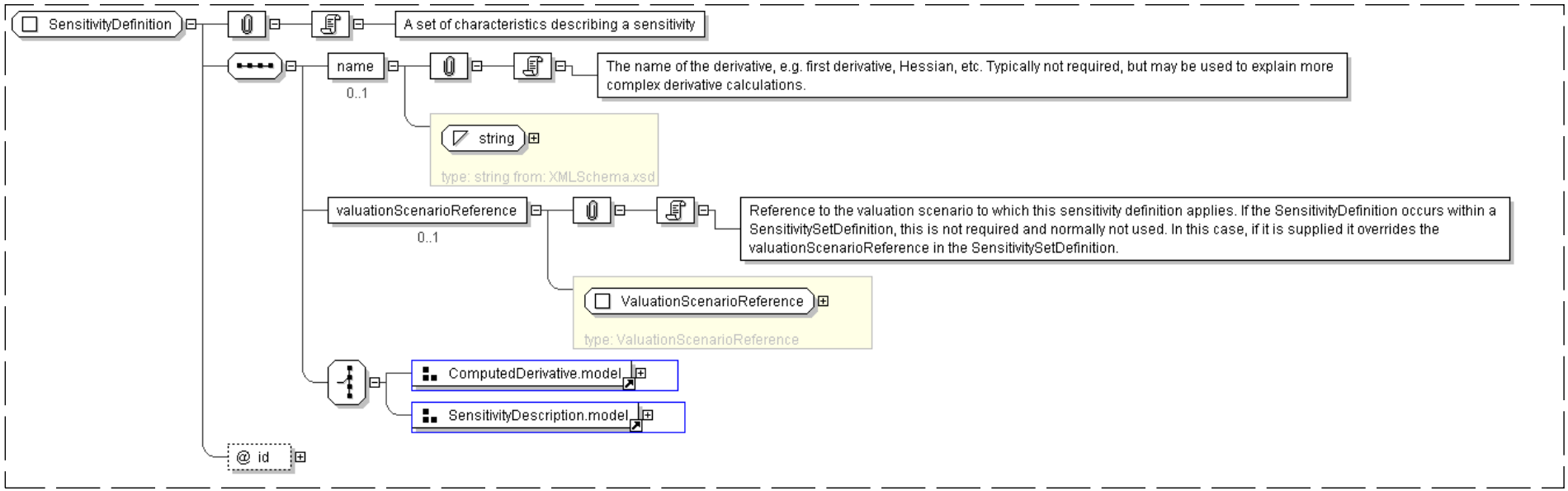
Name	SensitivityDefinition
Used by (from the same schema document)	Complex Type **SensitivitySetDefinition**
Abstract	no
Documentation	A set of characteristics describing a sensitivity

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>
```

top

Complex Type: SensitivitySetDefinition

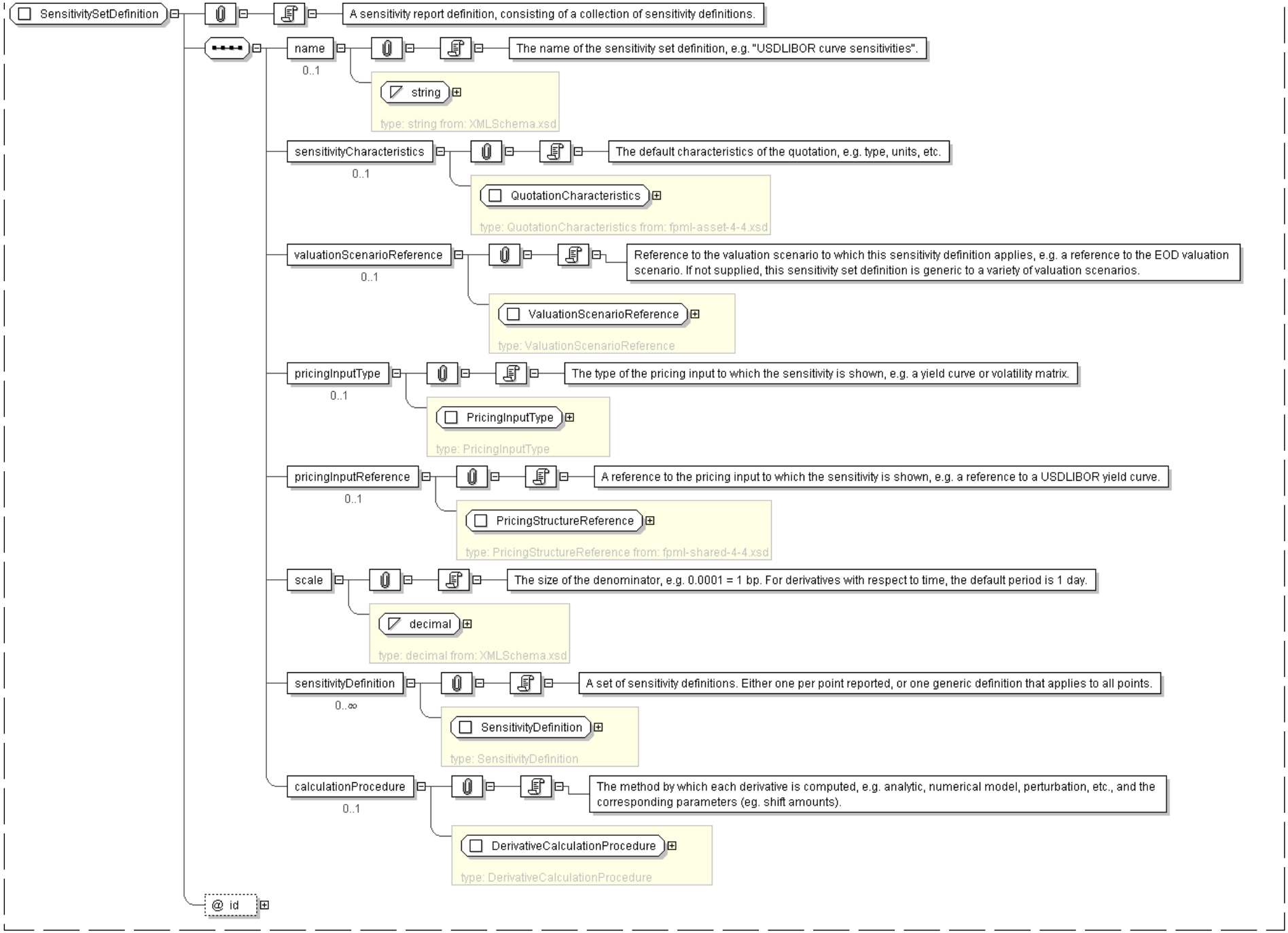
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: TimeDimension

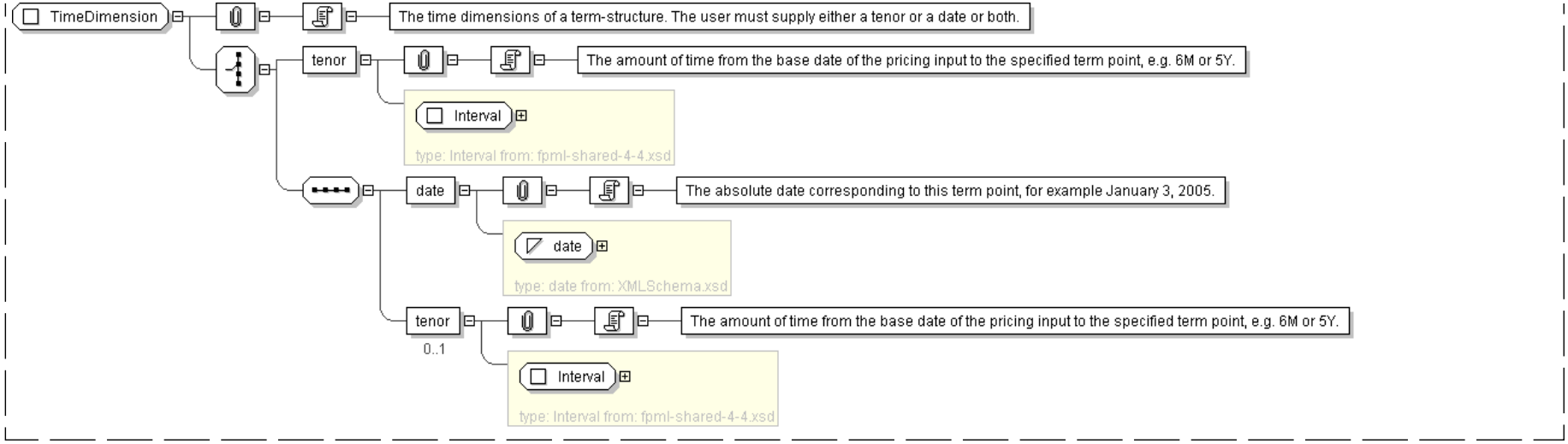
Super-types:	None
Sub-types:	None

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> Interval </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> Interval </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=" Interval "/>
      <xsd:sequence>
         <xsd:element name="date" type=" xsd:date "/>
         <xsd:element name="tenor" type=" Interval " minOccurs="0"/>
      </xsd:sequence>
   </xsd:choice>
</xsd:complexType>
```

top

Complex Type: Valuation

Super-types:	None
Sub-types:	• BasicAssetValuation (by extension) • PricingStructureValuation (by extension)

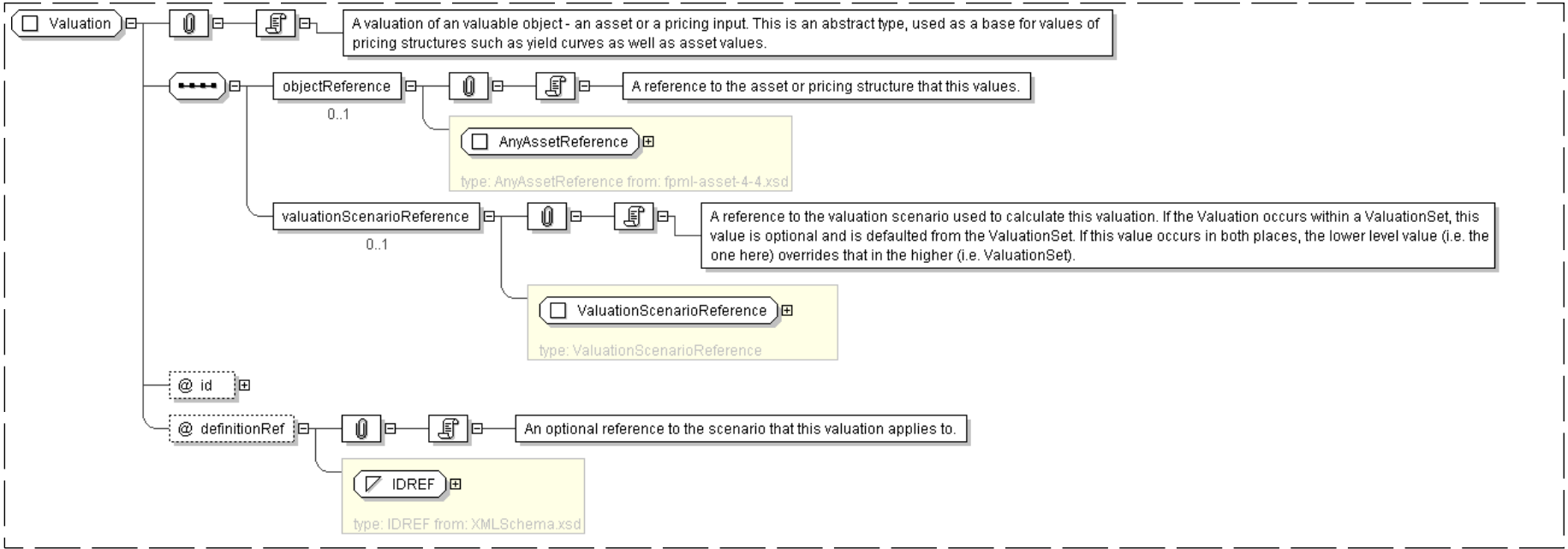
Name	Valuation
Abstract	no
Documentation	A valuation of an valuable object - an asset or a pricing input. This is an abstract type, used as a base for values of pricing structures such as yield curves as well as asset values.

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>
```

top

Complex Type: ValuationReference

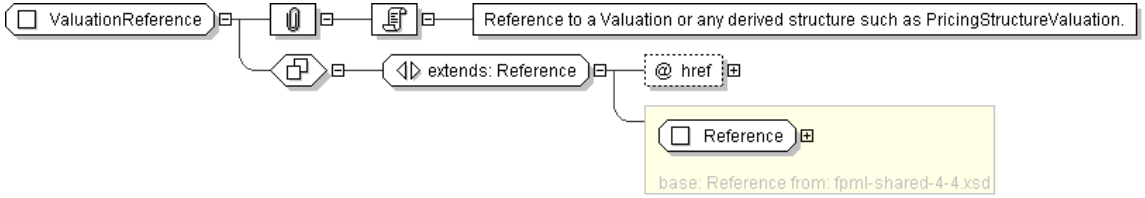
Super-types:	Reference < **ValuationReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: ValuationScenario

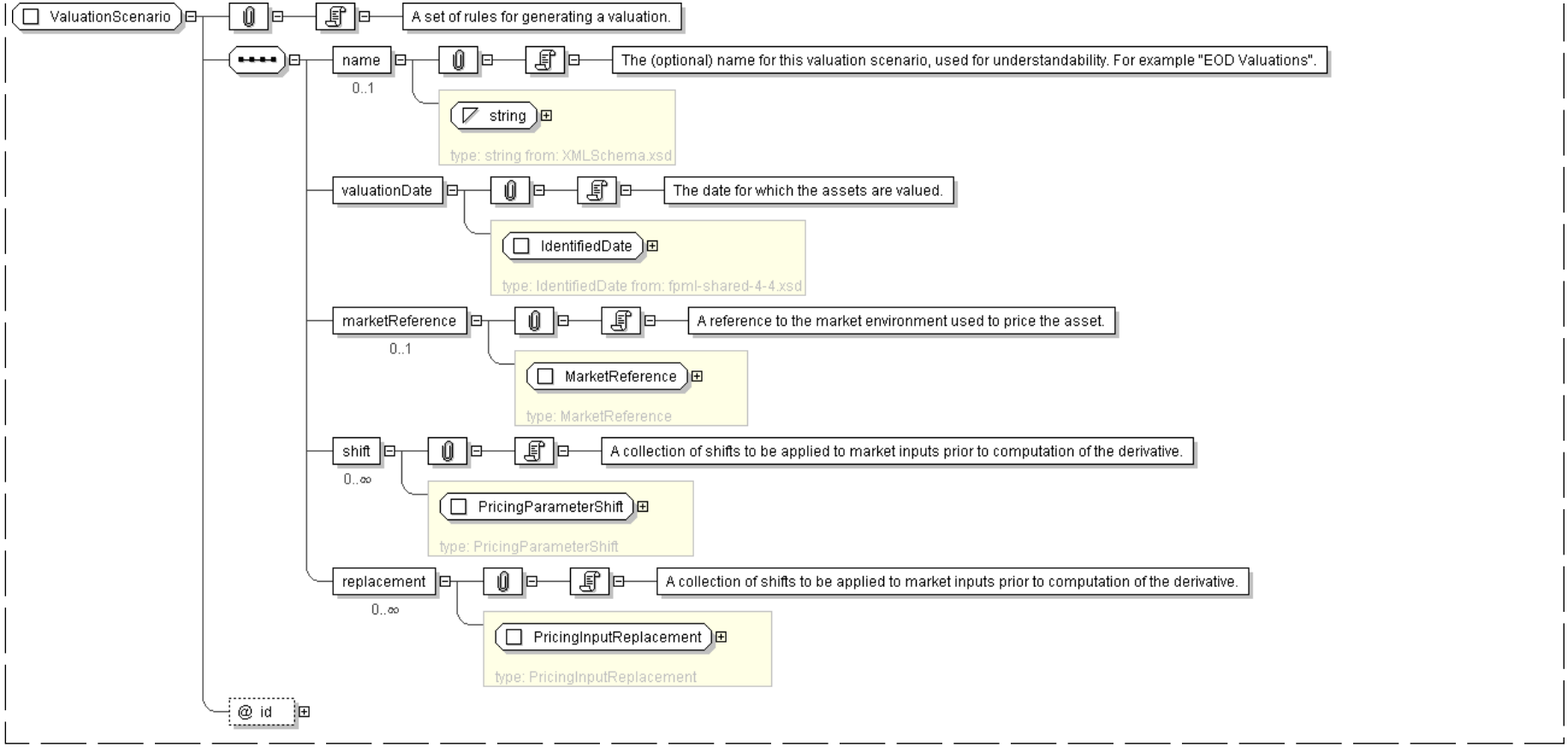
Super-types:	None
Sub-types:	None

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>
```

top

Complex Type: ValuationScenarioReference

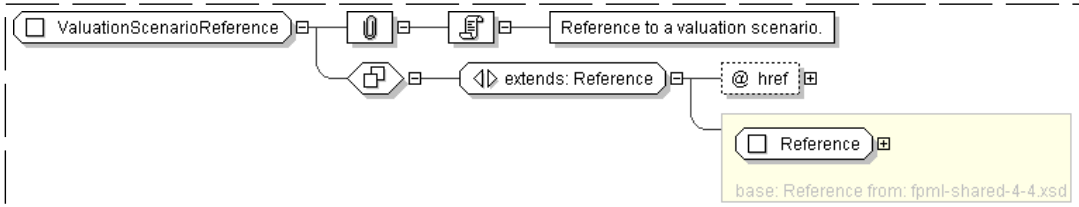
Super-types:	Reference < **ValuationScenarioReference** (by extension)
Sub-types:	None

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>
```

top

Complex Type: WeightedPartialDerivative

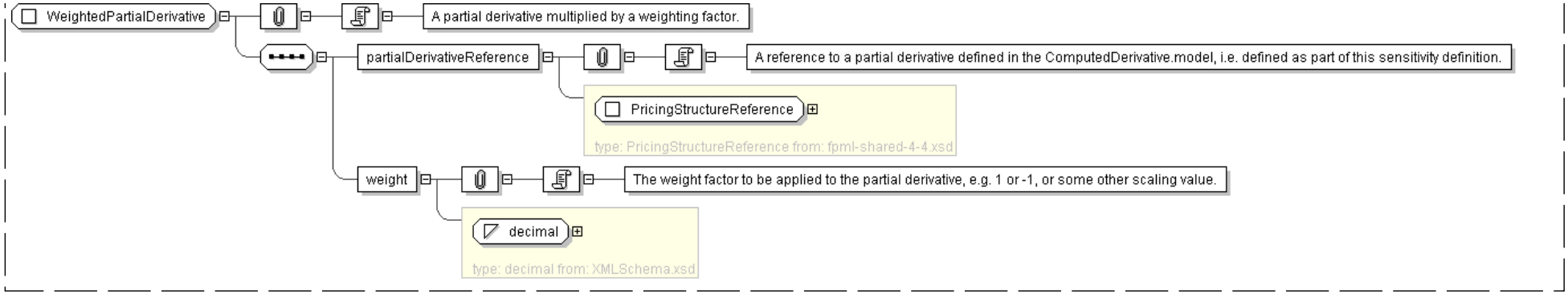
Super-types:	None
Sub-types:	None

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>
```

top

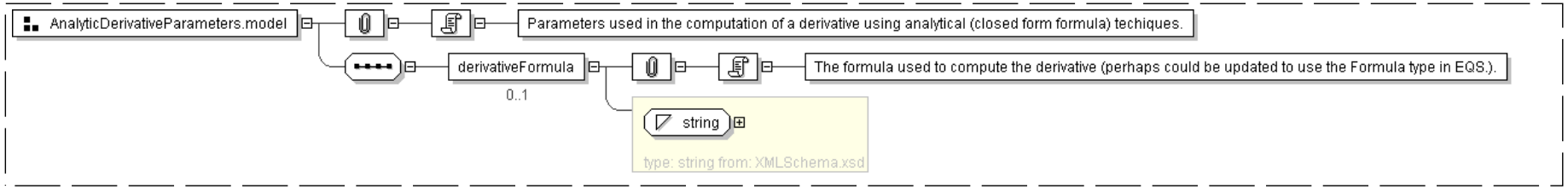
Model Group: AnalyticDerivativeParameters.model

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>
```

top

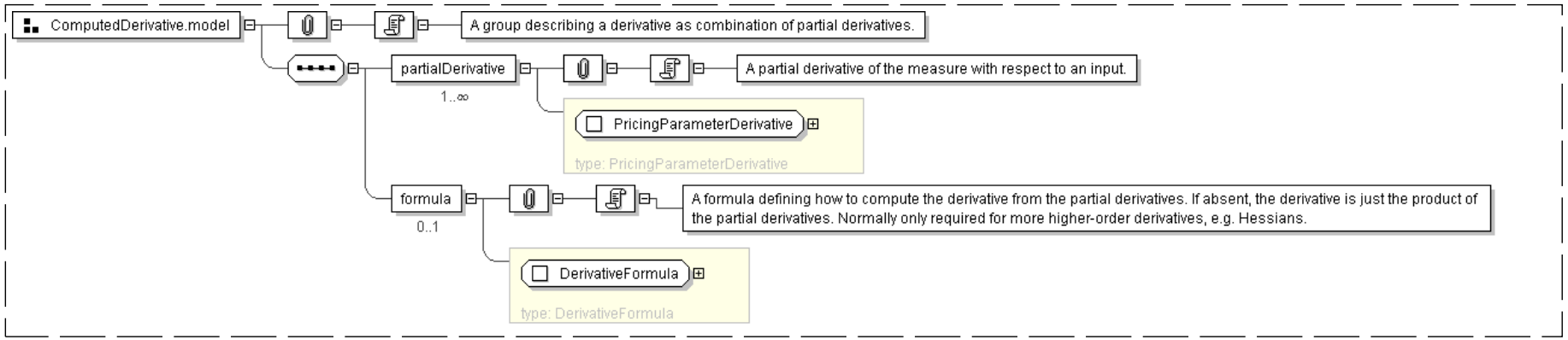
Model Group: ComputedDerivative.model

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>
```

top

Model Group: DerivativeCalculationParameters.model

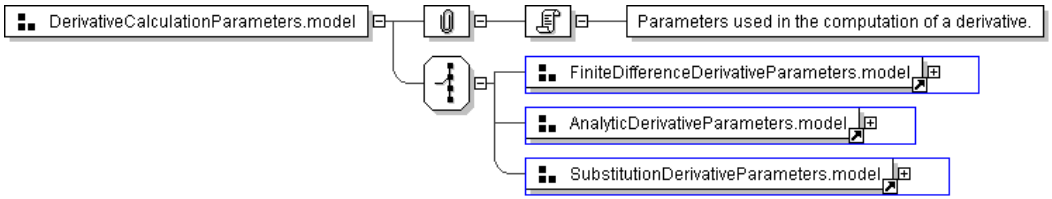
Name	DerivativeCalculationParameters.model
Used by (from the same schema document)	Complex Type **DerivativeCalculationProcedure**
Documentation	Parameters used in the computation of a derivative.

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>
```

top

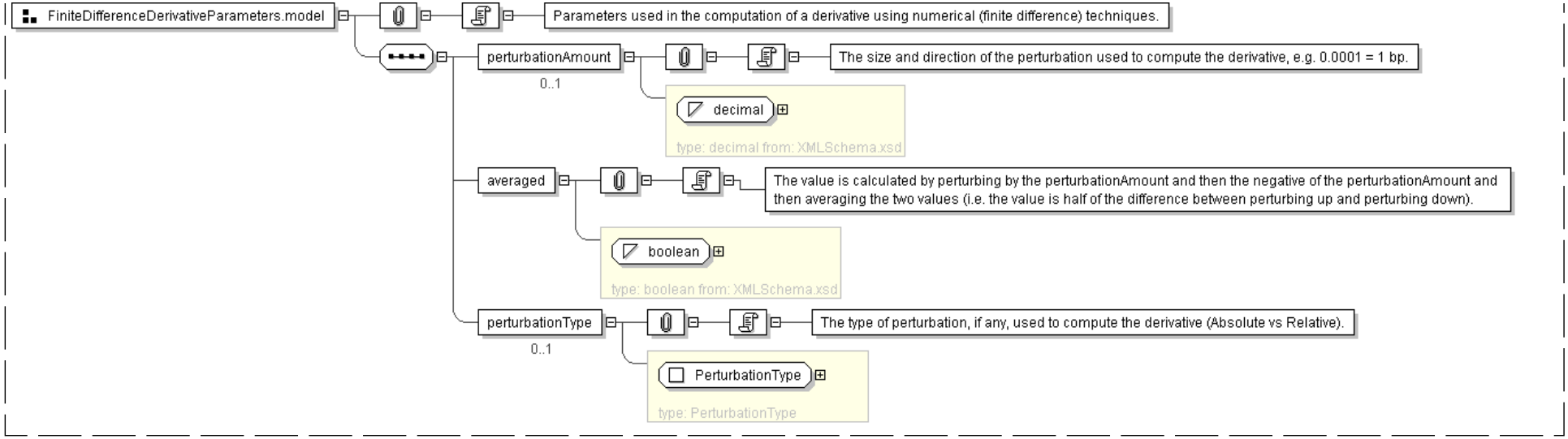
Model Group: FiniteDifferenceDerivativeParameters.model

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>
```

top

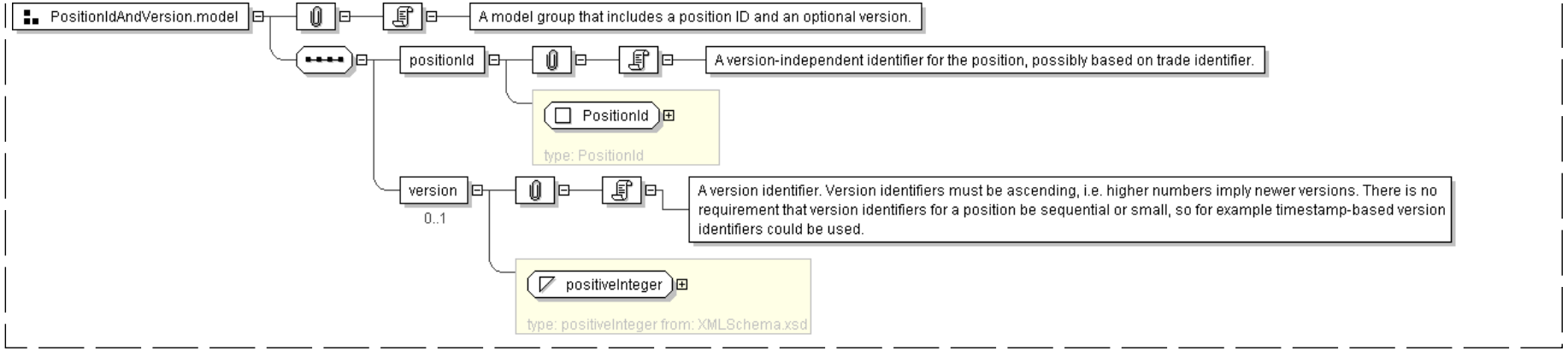
Model Group: PositionIdAndVersion.model

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>
```

top

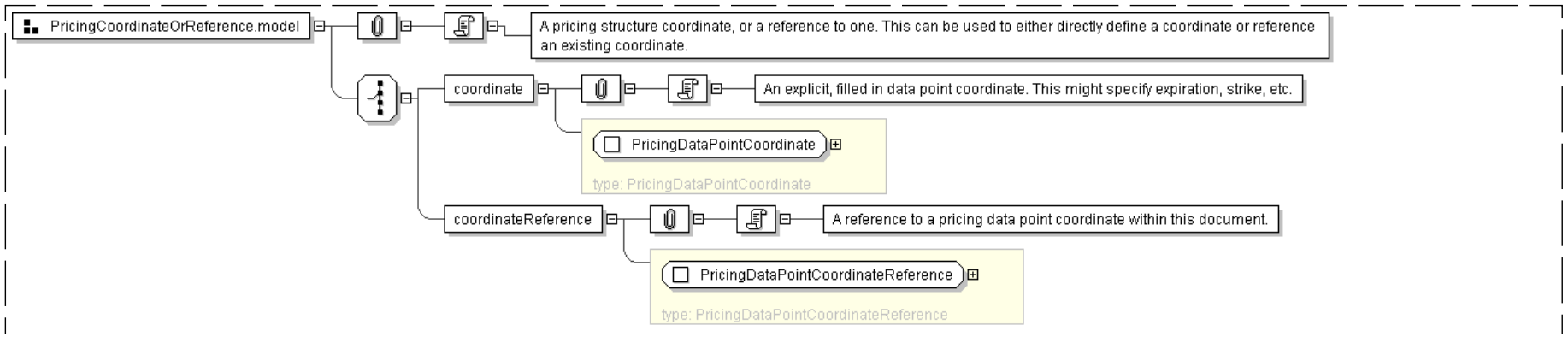
Model Group: PricingCoordinateOrReference.model

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>
```

top

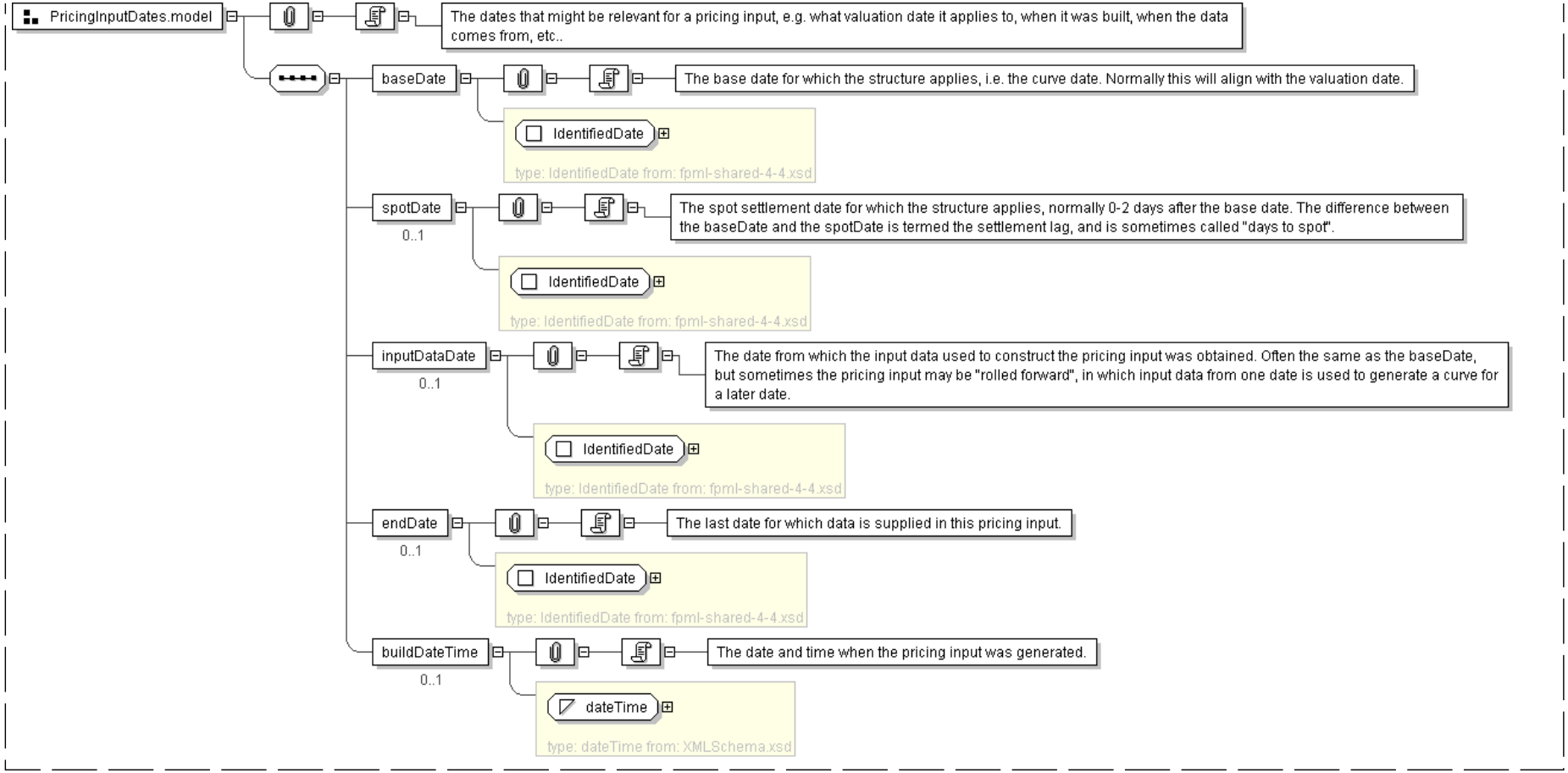
Model Group: PricingInputDates.model

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>
```

top

Model Group: PricingStructureIndex.model

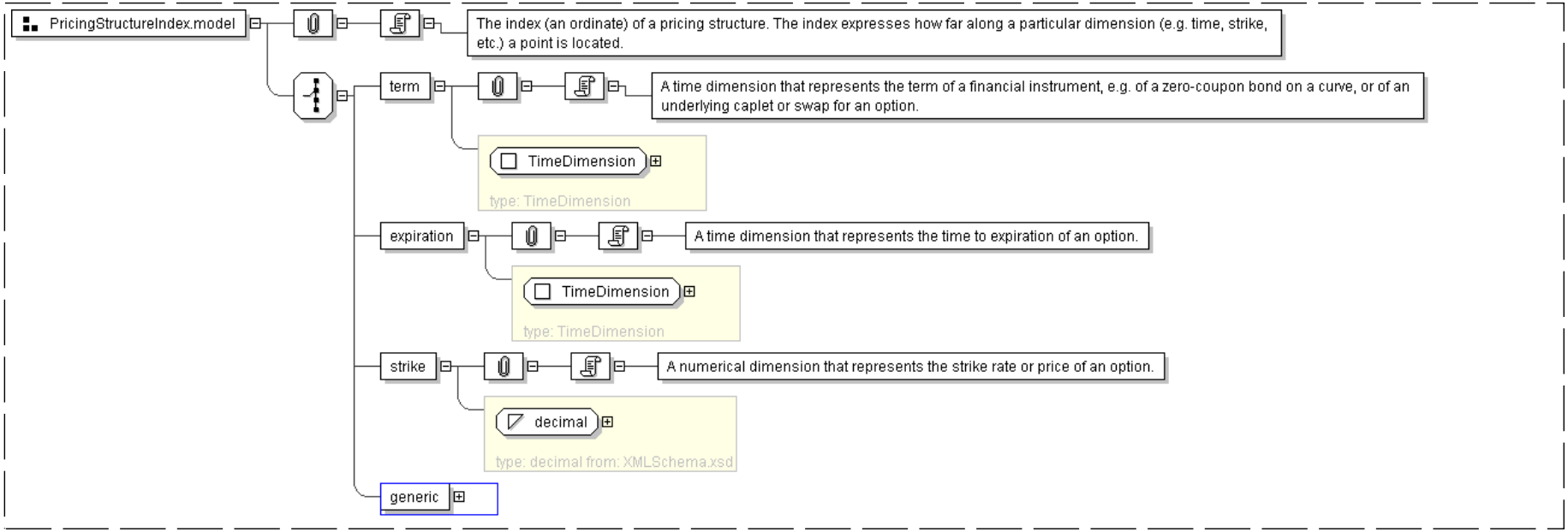
Name	PricingStructureIndex.model
Used by (from the same schema document)	Complex Type **PricingDataPointCoordinate**
Documentation	The index (an ordinate) of a pricing structure. The index expresses how far along a particular dimension (e.g. time, strike, etc.) a point is located.

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>
```

top

Model Group: SensitivityDescription.model

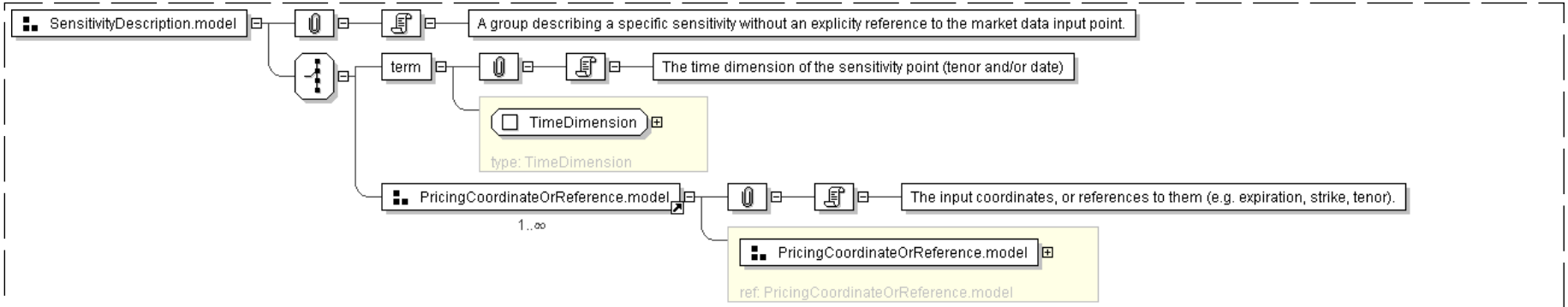
Name	SensitivityDescription.model
Used by (from the same schema document)	Complex Type **SensitivityDefinition**
Documentation	A group describing a specific sensitivity without an explicity reference to the market data input point.

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>
```

top

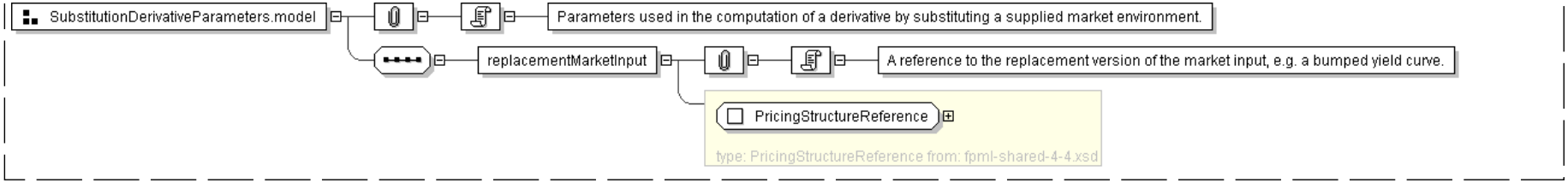
Model Group: SubstitutionDerivativeParameters.model

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>
```

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.