XML Schema Documentation

Table of Contents

- Schema Document Properties
- Global Declarations
  - Element: **interestLeg**
  - Element: **returnLeg**
  - Element: **returnSwap**
  - Element: **returnSwapLeg**
  - Element: **varianceLeg**
- Global Definitions
  - Complex Type: **AdditionalDisruptionEvents**
  - Complex Type: **AdditionalPaymentAmount**
  - Complex Type: **AdjustableDateOrRelativeDateSequence**
  - Complex Type: **BoundedCorrelation**
  - Complex Type: **BoundedVariance**
  - Complex Type: **CalculatedAmount**
  - Complex Type: **CalculationFromObservation**
  - Complex Type: **Compounding**
  - Complex Type: **CompoundingRate**
  - Complex Type: **Correlation**
  - Complex Type: **DeprecatedVariance**
  - Complex Type: **DeprecatedVarianceAmount**
  - Complex Type: **DeprecatedVarianceLeg**
  - Complex Type: **DirectionalLeg**
  - Complex Type: **DirectionalLegUnderlyer**
  - Complex Type: **DirectionalLegUnderlyerValuation**
  - Complex Type: **DividendAdjustment**
  - Complex Type: **DividendPeriod**
  - Complex Type: **DividendPeriodDividend**
  - Complex Type: **EquityCorporateEvents**
  - Complex Type: **EquityPremium**
  - Complex Type: **EquityStrike**
  - Complex Type: **EquityValuation**
  - Complex Type: **ExtraordinaryEvents**
  - Complex Type: **IndexAdjustmentEvents**
  - Complex Type: **InterestCalculation**
  - Complex Type: **InterestCalculationReference**
  - Complex Type: **InterestLeg**
  - Complex Type: **InterestLegCalculationPeriodDates**
  - Complex Type: **InterestLegCalculationPeriodDatesReference**
  - Complex Type: **InterestLegResetDates**
  - Complex Type: **LegAmount**
  - Complex Type: **MakeWholeProvisions**
  - Complex Type: **NettedSwapBase**
  - Complex Type: **OptionFeatures**
  - Complex Type: **PrincipalExchangeAmount**
  - Complex Type: **PrincipalExchangeDescriptions**
  - Complex Type: **PrincipalExchangeFeatures**
  - Complex Type: **Representations**
  - Complex Type: **Return**
  - Complex Type: **ReturnLeg**
  - Complex Type: **ReturnLegValuation**
  - Complex Type: **ReturnLegValuationPrice**
  - Complex Type: **ReturnSwap**
  - Complex Type: **ReturnSwapAdditionalPayment**
  - Complex Type: **ReturnSwapAmount**
  - Complex Type: **ReturnSwapBase**

- Complex Type: **ReturnSwapEarlyTermination**
- Complex Type: **ReturnSwapLeg**
- Complex Type: **ReturnSwapLegUnderlyer**
- Complex Type: **ReturnSwapNotional**
- Complex Type: **ReturnSwapPaymentDates**
- Complex Type: **StartingDate**
- Complex Type: **StubCalculationPeriod**
- Complex Type: **Variance**
- Model Group: **Feature.model**

- Legend
- Glossary

top

Schema Document Properties

Target Namespace	**http://www.fpml.org/2007/FpML-4-4**
Version	$Revision: 2654 $
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Schema Composition	• This schema includes components from the following schema document(s): ○ fpml-option-shared-4-4.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: 2654 $"
attributeFormDefault="unqualified" elementFormDefault="qualified">
   <xsd:include schemaLocation="fpml-option-shared-4-4.xsd"/>
   ...
</xsd:schema>
```

top

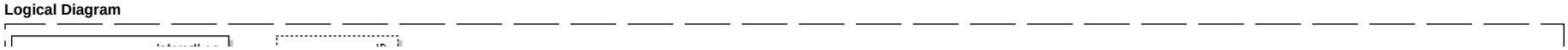
Global Declarations

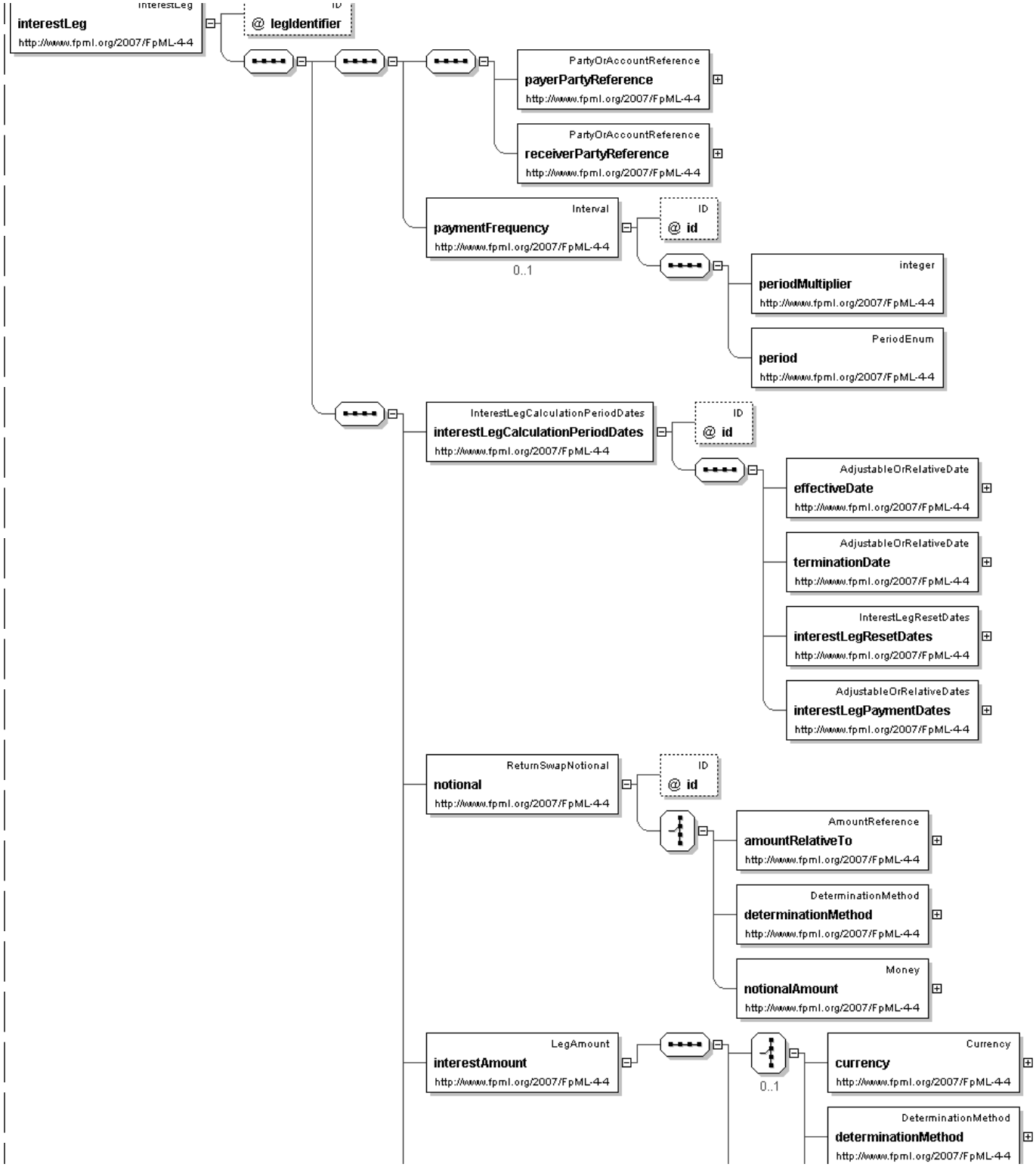
Element: interestLeg

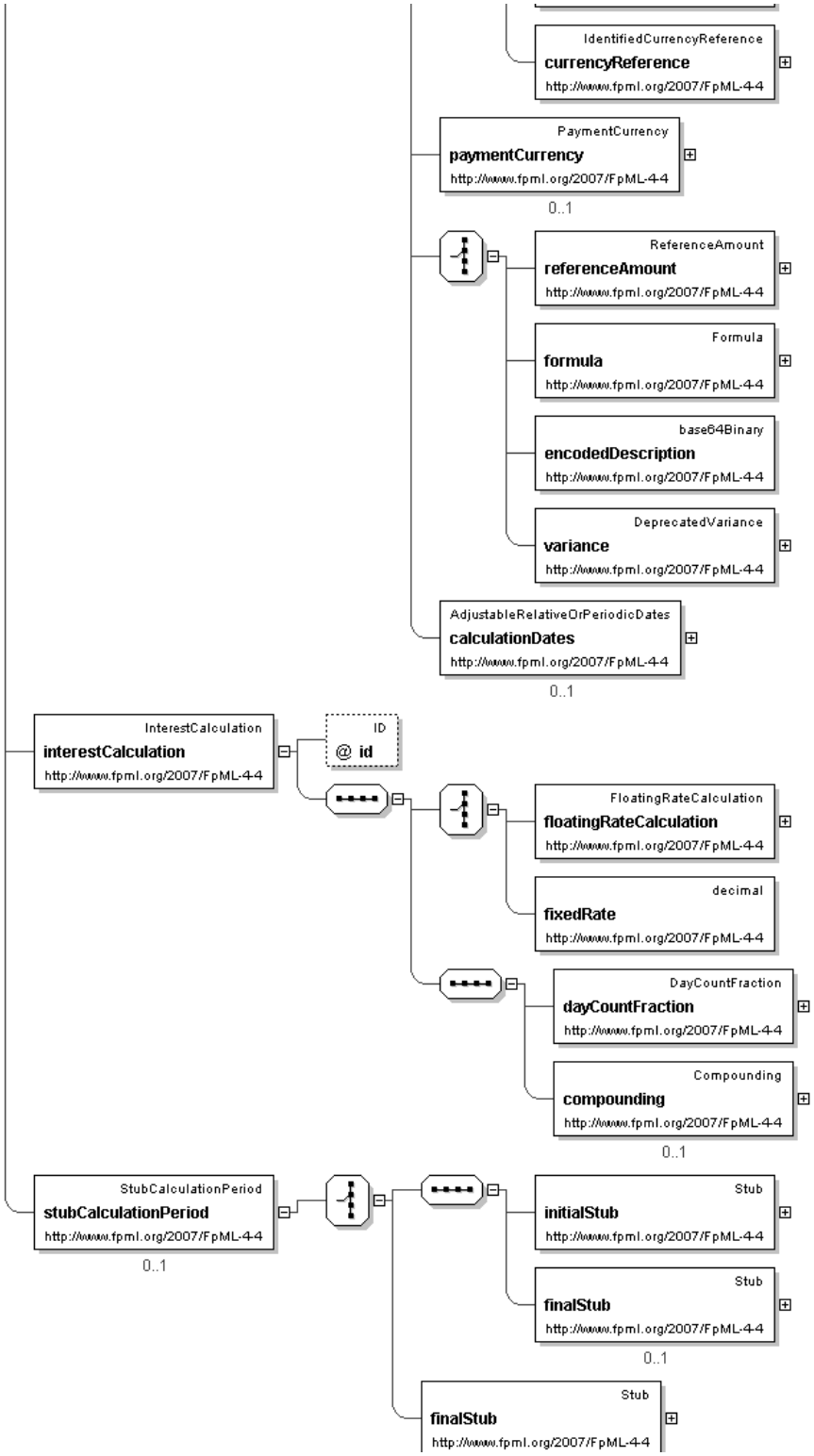
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	interestLeg
Type	InterestLeg
Nillable	no
Abstract	no
Documentation	The fixed income amounts of the return type swap.

Logical Diagram



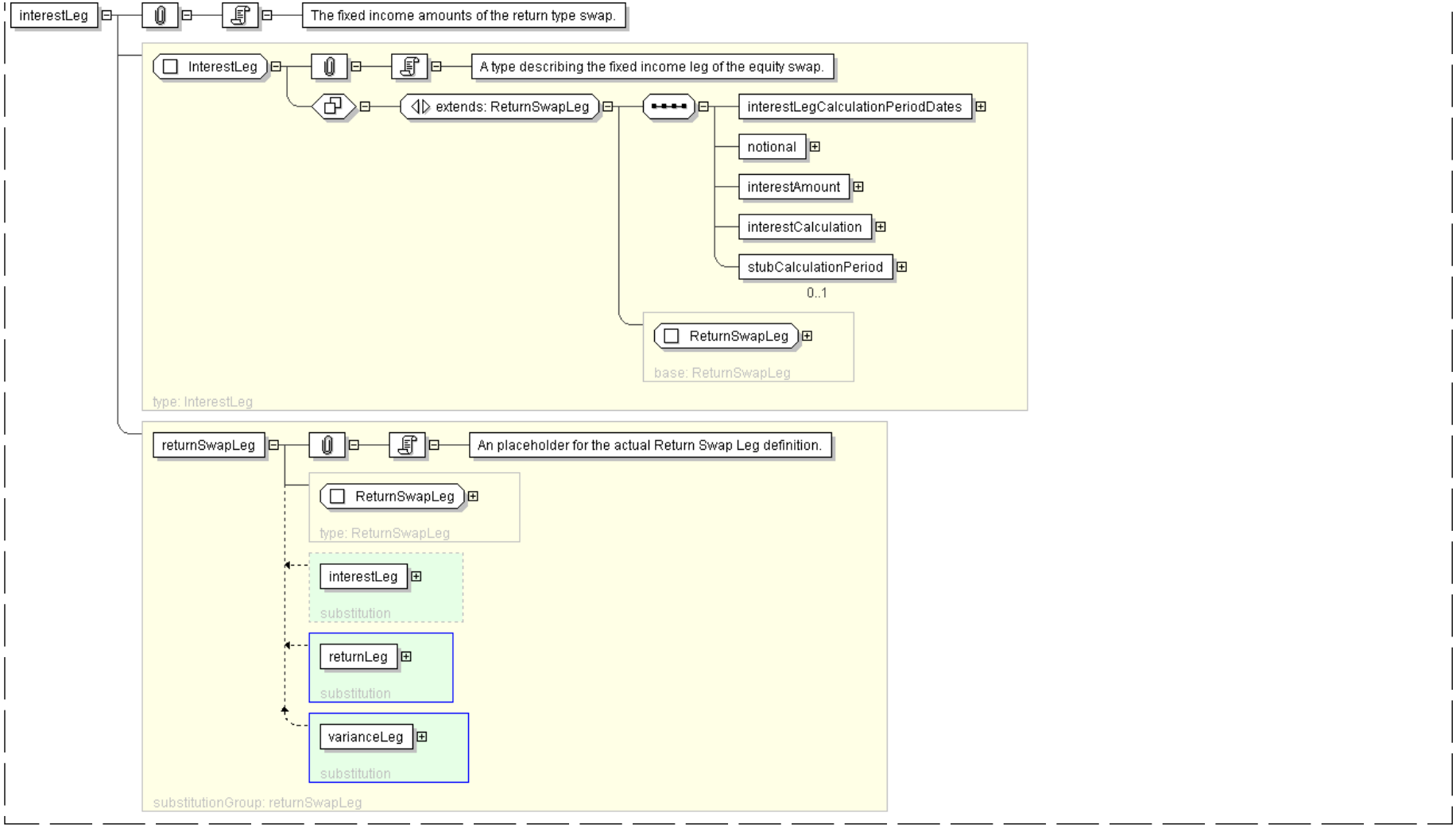

InterestLeg
interestLeg
http://www.fpml.org/2007/FpML-4-4
ID
@ legIdentifier
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2007/FpML-4-4
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2007/FpML-4-4
Interval
paymentFrequency
http://www.fpml.org/2007/FpML-4-4
0..1
ID
@ id
integer
periodMultiplier
http://www.fpml.org/2007/FpML-4-4
PeriodEnum
period
http://www.fpml.org/2007/FpML-4-4
InterestLegCalculationPeriodDates
interestLegCalculationPeriodDates
http://www.fpml.org/2007/FpML-4-4
ID
@ id
AdjustableOrRelativeDate
effectiveDate
http://www.fpml.org/2007/FpML-4-4
AdjustableOrRelativeDate
terminationDate
http://www.fpml.org/2007/FpML-4-4
InterestLegResetDates
interestLegResetDates
http://www.fpml.org/2007/FpML-4-4
AdjustableOrRelativeDates
interestLegPaymentDates
http://www.fpml.org/2007/FpML-4-4
ReturnSwapNotional
notional
http://www.fpml.org/2007/FpML-4-4
ID
@ id
AmountReference
amountRelativeTo
http://www.fpml.org/2007/FpML-4-4
DeterminationMethod
determinationMethod
http://www.fpml.org/2007/FpML-4-4
Money
notionalAmount
http://www.fpml.org/2007/FpML-4-4
LegAmount
interestAmount
http://www.fpml.org/2007/FpML-4-4
0..1
Currency
currency
http://www.fpml.org/2007/FpML-4-4
DeterminationMethod
determinationMethod
http://www.fpml.org/2007/FpML-4-4


IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2007/FpML-4-4
PaymentCurrency
paymentCurrency
http://www.fpml.org/2007/FpML-4-4
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2007/FpML-4-4
Formula
formula
http://www.fpml.org/2007/FpML-4-4
base64Binary
encodedDescription
http://www.fpml.org/2007/FpML-4-4
DeprecatedVariance
variance
http://www.fpml.org/2007/FpML-4-4
AdjustableRelativeOrPeriodicDates
calculationDates
http://www.fpml.org/2007/FpML-4-4
0..1
InterestCalculation
interestCalculation
http://www.fpml.org/2007/FpML-4-4
ID
@ id
FloatingRateCalculation
floatingRateCalculation
http://www.fpml.org/2007/FpML-4-4
decimal
fixedRate
http://www.fpml.org/2007/FpML-4-4
DayCountFraction
dayCountFraction
http://www.fpml.org/2007/FpML-4-4
Compounding
compounding
http://www.fpml.org/2007/FpML-4-4
0..1
StubCalculationPeriod
stubCalculationPeriod
http://www.fpml.org/2007/FpML-4-4
0..1
Stub
initialStub
http://www.fpml.org/2007/FpML-4-4
Stub
finalStub
http://www.fpml.org/2007/FpML-4-4
0..1
Stub
finalStub
http://www.fpml.org/2007/FpML-4-4

XML Instance Representation

```
<interestLeg
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

   <interestLegCalculationPeriodDates> InterestLegCalculationPeriodDates
   </interestLegCalculationPeriodDates> [1]
   'Component that holds the various dates used to specify the interest leg of the equity swap.
   It is used to define the InterestPeriodDates identifyer.'

   <notional> ReturnSwapNotional </notional> [1]
   'Specifies the notional of a return type swap. When used in the equity leg, the definition
   will typically combine the actual amount (using the notional component defined by the
   FpML industry group) and the determination method. When used in the interest leg,
   the definition will typically point to the definition of the equity leg.'

   <interestAmount> LegAmount </interestAmount> [1]
   'Specifies, in relation to each Interest Payment Date, the amount to which the Interest
   Payment Date relates. Unless otherwise specified, this term has the meaning defined in the
   ISDA 2000 ISDA Definitions.'

   <interestCalculation> InterestCalculation </interestCalculation> [1]
   'Specifies the calculation method of the interest rate leg of the equity swap. Includes
   the floating or fixed rate calculation definitions, along with the determination of the
   day count fraction.'

   <stubCalculationPeriod> StubCalculationPeriod </stubCalculationPeriod> [0..1]
   'Specifies the stub calculation period'

</interestLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="interestLeg" type=" InterestLeg " substitutionGroup="returnSwapLeg"/>
```

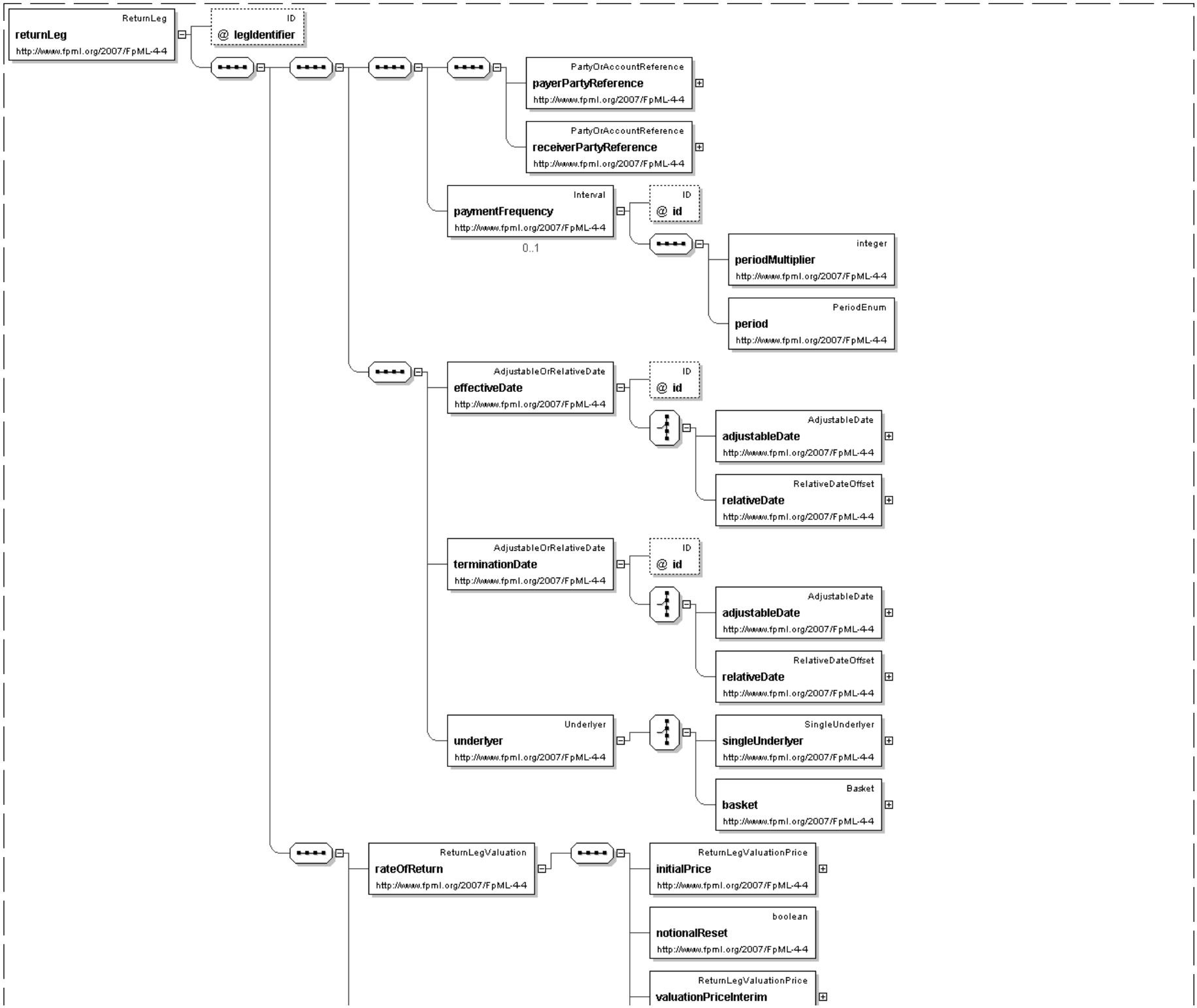
top

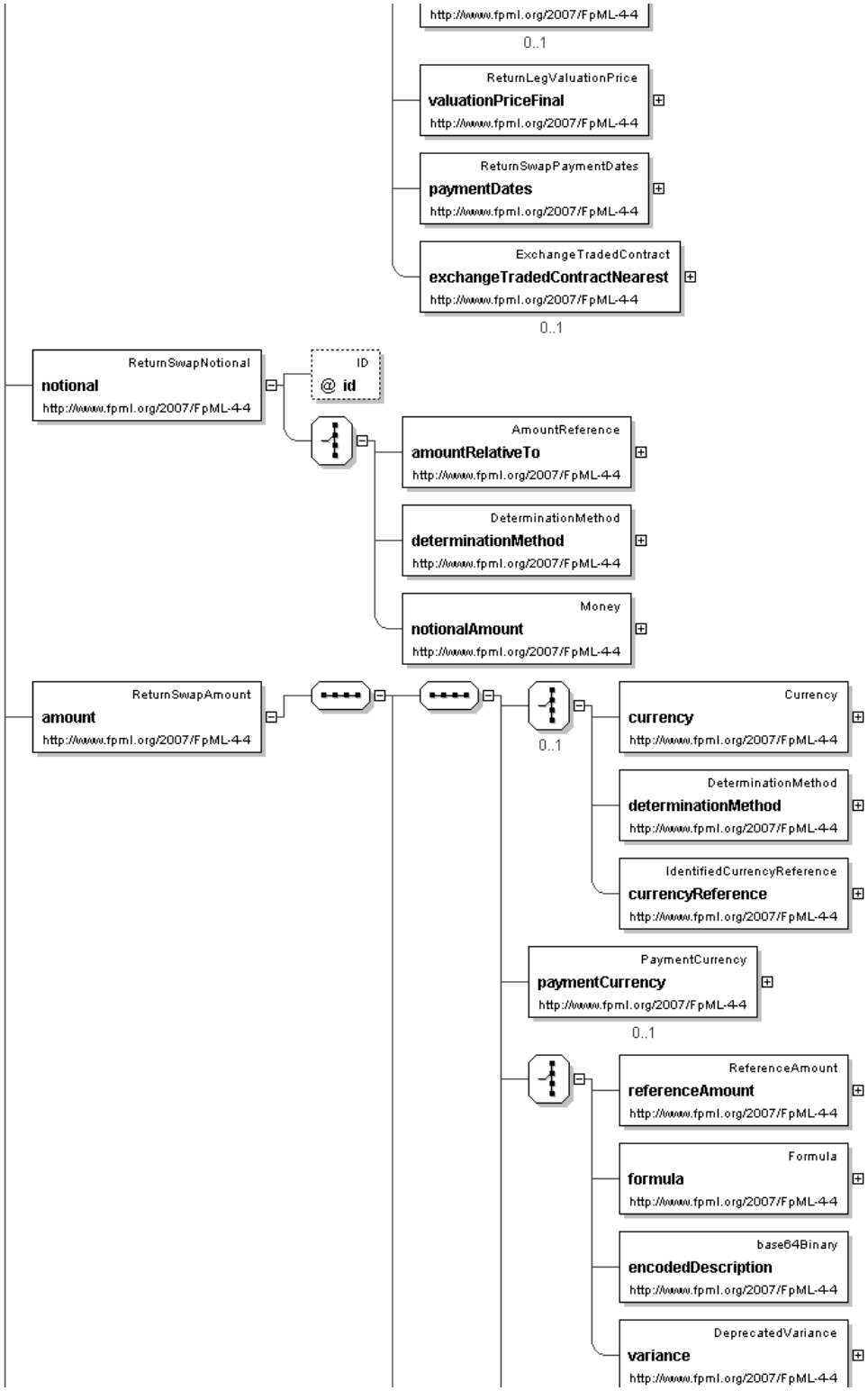
Element: returnLeg

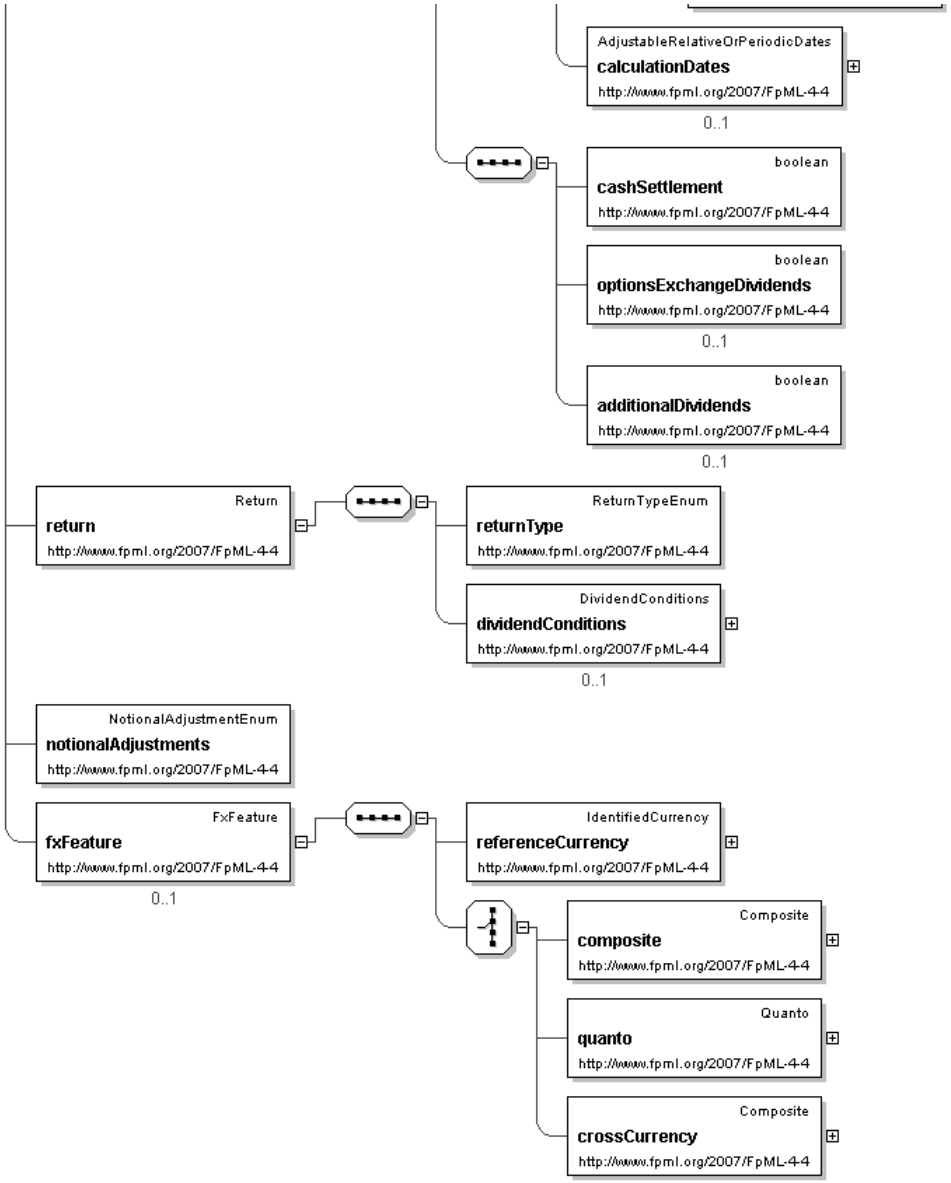
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	returnLeg
Type	ReturnLeg
Nillable	no
Abstract	no
Documentation	Return amounts of the return type swap.

Logical Diagram


ReturnLeg
returnLeg
http://www.fpml.org/2007/FpML-4-4
ID
@ legIdentifier
PartyOrAccountReference
payerPartyReference
http://www.fpml.org/2007/FpML-4-4
PartyOrAccountReference
receiverPartyReference
http://www.fpml.org/2007/FpML-4-4
Interval
paymentFrequency
http://www.fpml.org/2007/FpML-4-4
0..1
ID
@ id
integer
periodMultiplier
http://www.fpml.org/2007/FpML-4-4
PeriodEnum
period
http://www.fpml.org/2007/FpML-4-4
AdjustableOrRelativeDate
effectiveDate
http://www.fpml.org/2007/FpML-4-4
ID
@ id
AdjustableDate
adjustableDate
http://www.fpml.org/2007/FpML-4-4
RelativeDateOffset
relativeDate
http://www.fpml.org/2007/FpML-4-4
AdjustableOrRelativeDate
terminationDate
http://www.fpml.org/2007/FpML-4-4
ID
@ id
AdjustableDate
adjustableDate
http://www.fpml.org/2007/FpML-4-4
RelativeDateOffset
relativeDate
http://www.fpml.org/2007/FpML-4-4
Underlyer
underlyer
http://www.fpml.org/2007/FpML-4-4
SingleUnderlyer
singleUnderlyer
http://www.fpml.org/2007/FpML-4-4
Basket
basket
http://www.fpml.org/2007/FpML-4-4
ReturnLegValuation
rateOfReturn
http://www.fpml.org/2007/FpML-4-4
ReturnLegValuationPrice
initialPrice
http://www.fpml.org/2007/FpML-4-4
boolean
notionalReset
http://www.fpml.org/2007/FpML-4-4
ReturnLegValuationPrice
valuationPriceInterim


http://www.fpml.org/2007/FpML-4-4
0..1
ReturnLegValuationPrice
valuationPriceFinal
http://www.fpml.org/2007/FpML-4-4
ReturnSwapPaymentDates
paymentDates
http://www.fpml.org/2007/FpML-4-4
ExchangeTradedContract
exchangeTradedContractNearest
http://www.fpml.org/2007/FpML-4-4
0..1
ReturnSwapNotional
notional
http://www.fpml.org/2007/FpML-4-4
ID
@ id
AmountReference
amountRelativeTo
http://www.fpml.org/2007/FpML-4-4
DeterminationMethod
determinationMethod
http://www.fpml.org/2007/FpML-4-4
Money
notionalAmount
http://www.fpml.org/2007/FpML-4-4
ReturnSwapAmount
amount
http://www.fpml.org/2007/FpML-4-4
0..1
Currency
currency
http://www.fpml.org/2007/FpML-4-4
DeterminationMethod
determinationMethod
http://www.fpml.org/2007/FpML-4-4
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2007/FpML-4-4
PaymentCurrency
paymentCurrency
http://www.fpml.org/2007/FpML-4-4
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2007/FpML-4-4
Formula
formula
http://www.fpml.org/2007/FpML-4-4
base64Binary
encodedDescription
http://www.fpml.org/2007/FpML-4-4
DeprecatedVariance
variance
http://www.fpml.org/2007/FpML-4-4



XML Instance Representation

```
<returnLeg
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'
```

<paymentFrequency> Interval </paymentFrequency> [0..1]

'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

<effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]

'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere else in the document (through the relativeDate component), this element will typically point to the effective date of the other leg of the swap.'

<terminationDate> AdjustableOrRelativeDate </terminationDate> [1]

'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere else in the document (through the relativeDate component), this element will typically point to the termination date of the other leg of the swap.'

<underlyer> Underlyer </underlyer> [1]

'Specifies the underlying component of the leg, which can be either one or many and consists in either equity, index or convertible bond component, or a combination of these.'

<rateOfReturn> ReturnLegValuation </rateOfReturn> [1]

'Element named \"valuation\" in versions prior to FpML 4.2 Second Working Draft. Specifies the terms of the initial price of the return type swap and of the subsequent valuations of the underlyer.'

<notional> ReturnSwapNotional </notional> [1]

'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine the actual amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest leg, the definition will typically point to the definition of the equity leg.'

<amount> ReturnSwapAmount </amount> [1]

'Element named \"equityAmount\" in versions prior to FpML 4.2 Second Working Draft. Specifies, in relation to each Payment Date, the amount to which the Payment Date relates. For equity swaps this element is equivalent to the Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.'

<return> Return </return> [1]

'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the amounts.'

<notionalAdjustments> NotionalAdjustmentEnum </notionalAdjustments> [1]

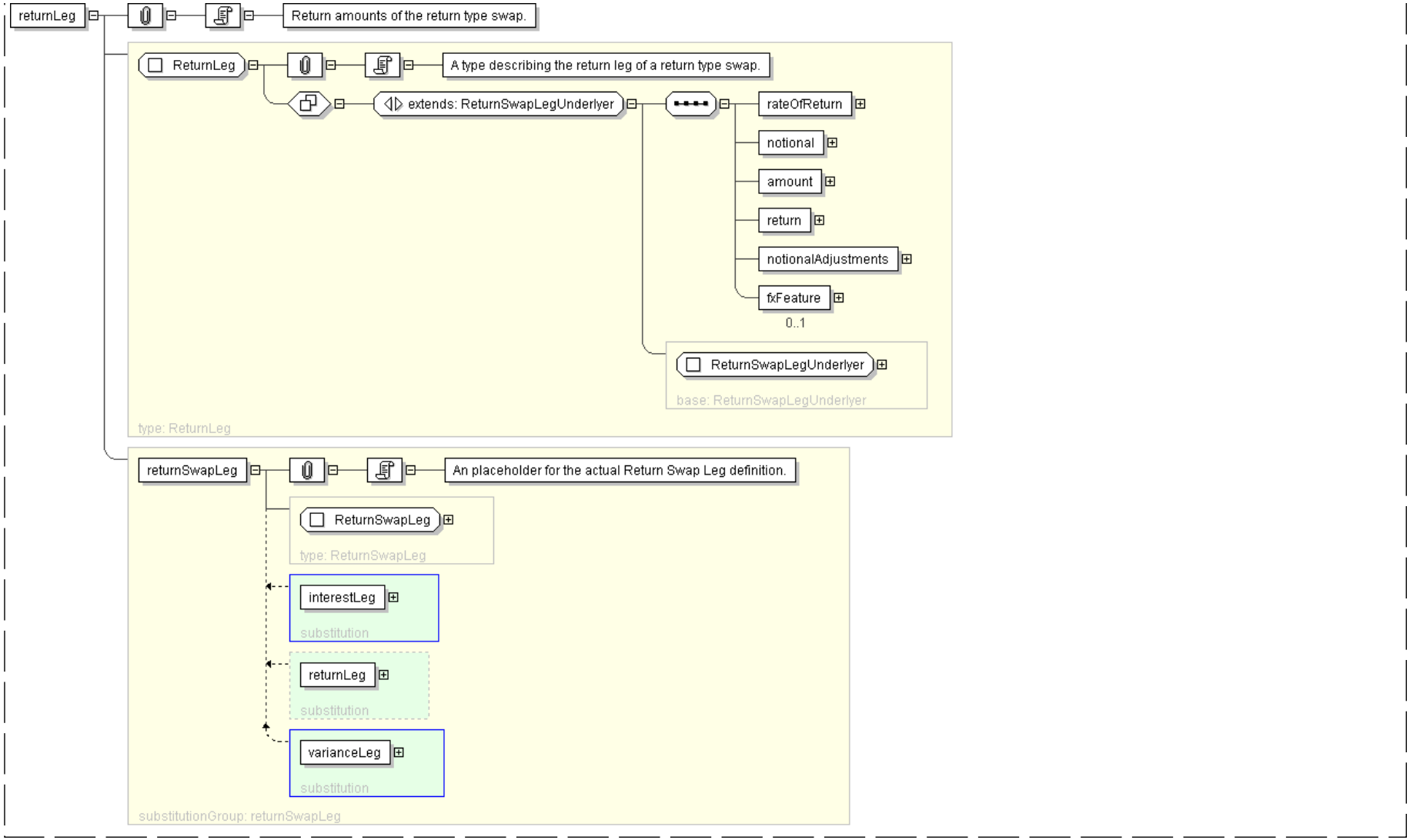
'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

<fxFeature> FxFeature </fxFeature> [0..1]

'A quanto or composite FX feature.'

</returnLeg>

Diagram



Schema Component Representation

```
<xsd:element name="returnLeg" type=" ReturnLeg " substitutionGroup="returnSwapLeg"/>
```

top

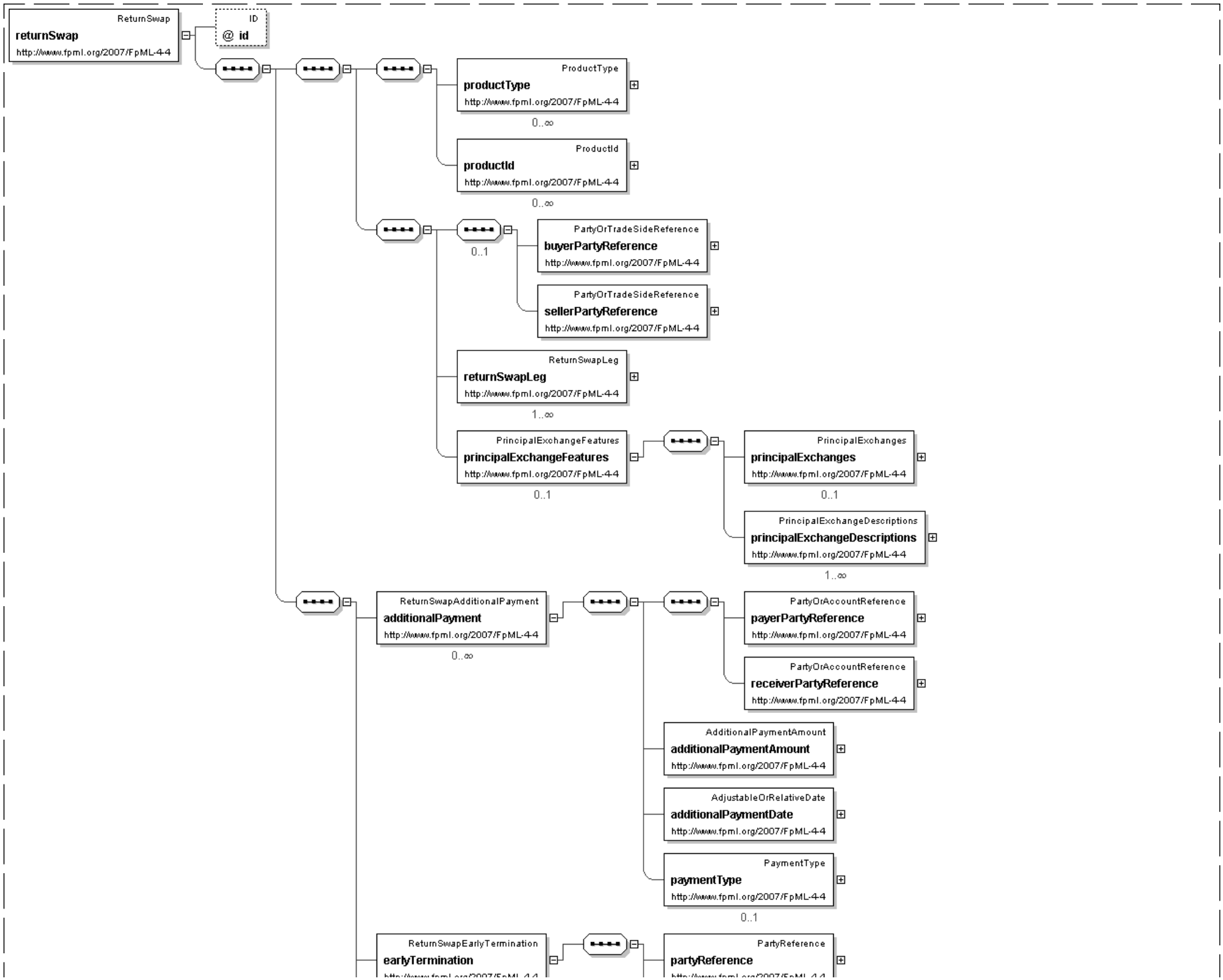
Element: returnSwap

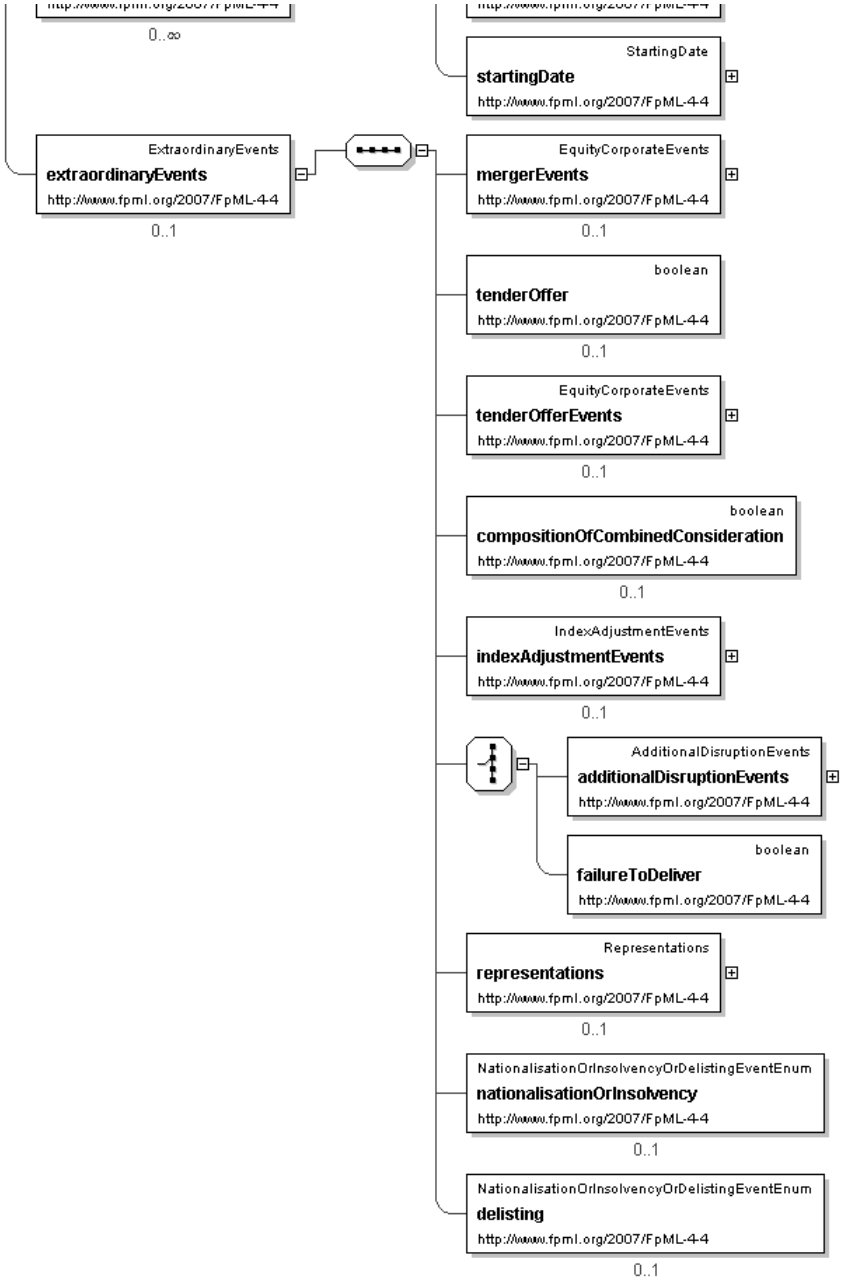
- *This element can be used wherever the following element is referenced:*
  - product

Name	returnSwap
Type	ReturnSwap
Nillable	no
Abstract	no

Documentation Specifies the structure of a return type swap. It can represent equity swaps, total return swaps, variance swaps.

Logical Diagram





XML Instance Representation

```
<returnSwap
id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using
   a coding scheme.'

   <productId> ProductId </productId> [0..*]
```

'A product reference identifier allocated by a party. FpML does not define the domain values associated with this element. Note that the domain values for this element are not strictly an enumerated list.'

Start Group: BuyerSeller.model [0..1]

'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support the situation where an implementor wishes to indicate who has manufactured the Swap through representing them as the Seller. It may be removed in future major revisions.'

<buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]

'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

<sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]

'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

End Group: BuyerSeller.model

<returnSwapLeg> ... </returnSwapLeg> [1..*]

<principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]

'This is used to document a Fully Funded Return Swap.'

<additionalPayment> ReturnSwapAdditionalPayment </additionalPayment> [0..*]

'Specifies additional payment(s) between the principal parties to the trade. This component extends some of the features of the additionalPayment component developed by the FpML industry group. Appropriate discussions will determine whether it would be appropriate to extend the shared component in order to meet the further requirements of equity swaps.'

<earlyTermination> ReturnSwapEarlyTermination </earlyTermination> [0..*]

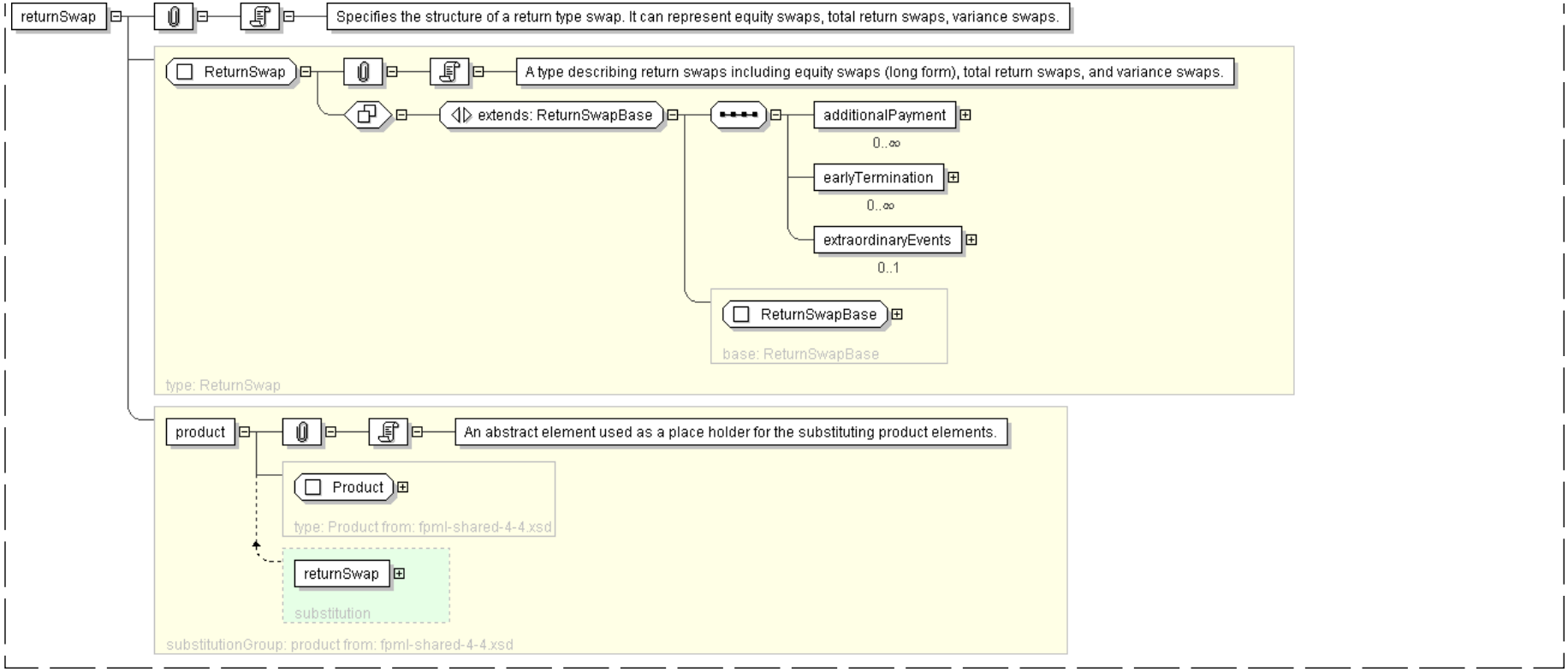
'Specifies, for one or for both the parties to the trade, the date from which it can early terminate it.'

<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]

'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.'

</returnSwap>

Diagram



Schema Component Representation

```
<xsd:element name="returnSwap" type=" ReturnSwap " substitutionGroup="product"/>
```

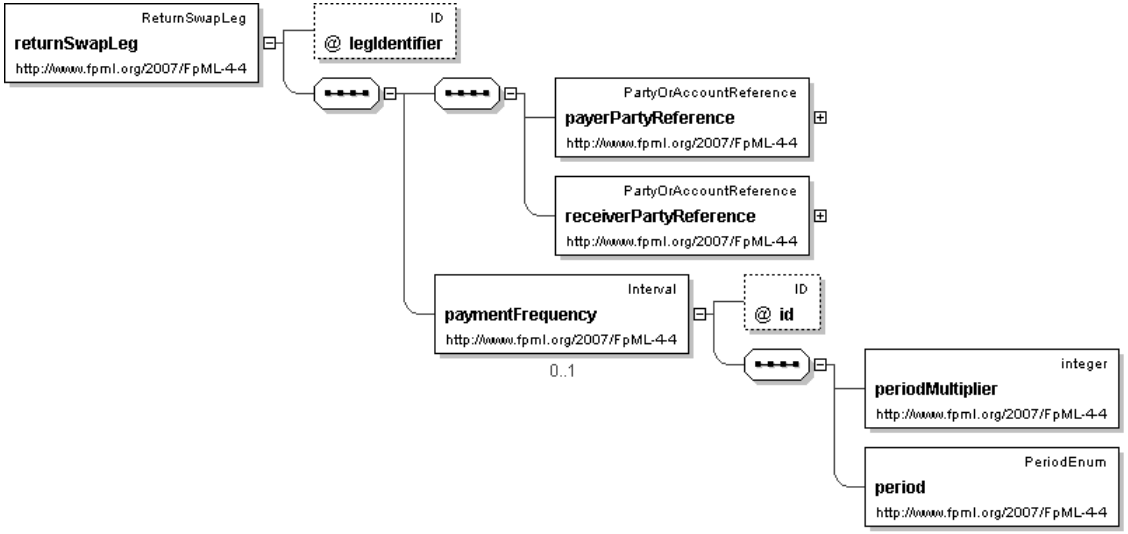
top

Element: returnSwapLeg

- *The following elements can be used wherever this element is referenced:*
  - interestLeg
  - returnLeg
  - varianceLeg

Name	returnSwapLeg
Used by (from the same schema document)	Complex Type **ReturnSwapBase**
Type	ReturnSwapLeg
Nillable	no
Abstract	yes
Documentation	An placeholder for the actual Return Swap Leg definition.

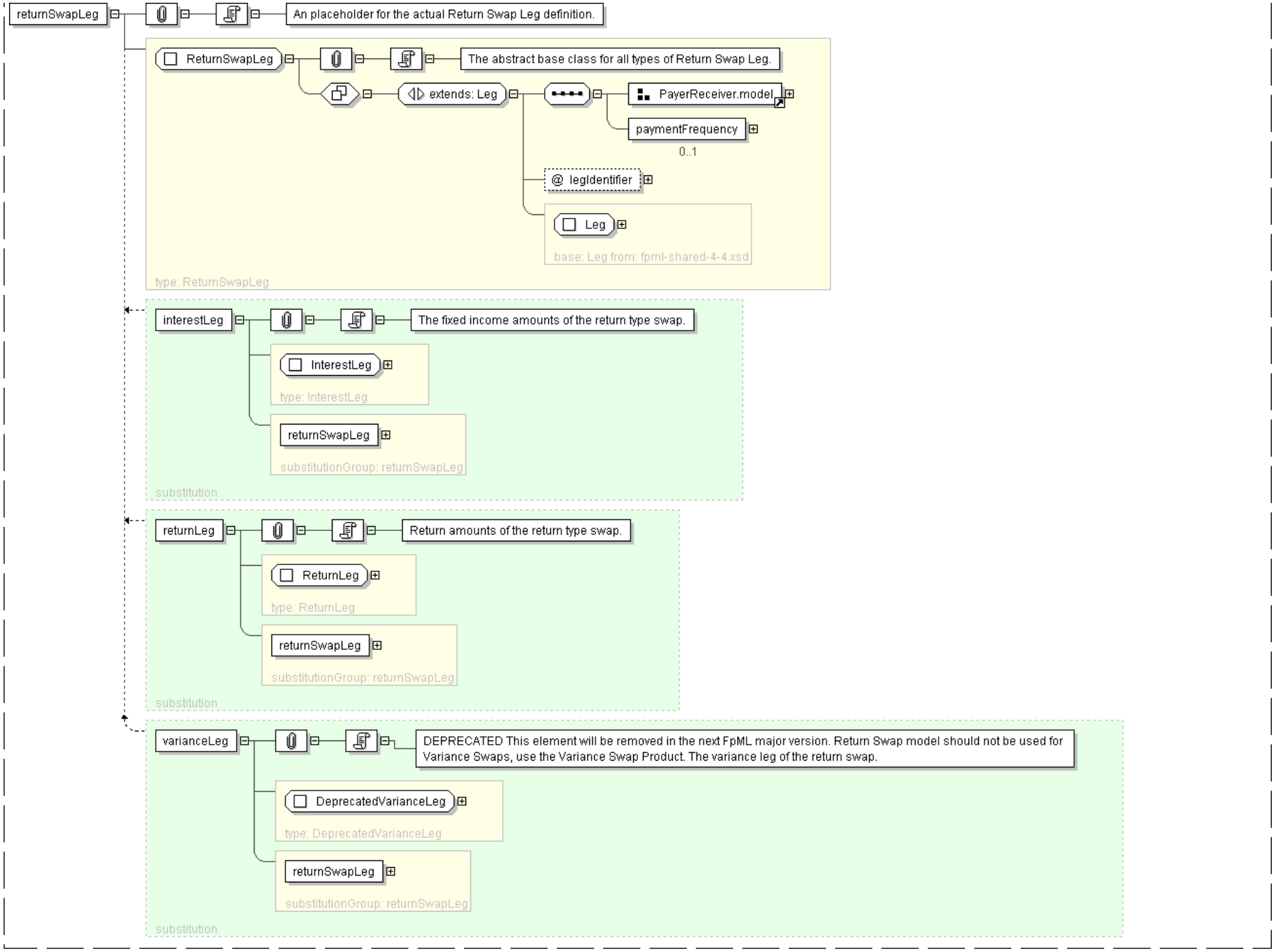
Logical Diagram



XML Instance Representation

```
<returnSwapLeg
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

</returnSwapLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="returnSwapLeg" type=" ReturnSwapLeg " abstract="true"/>
```

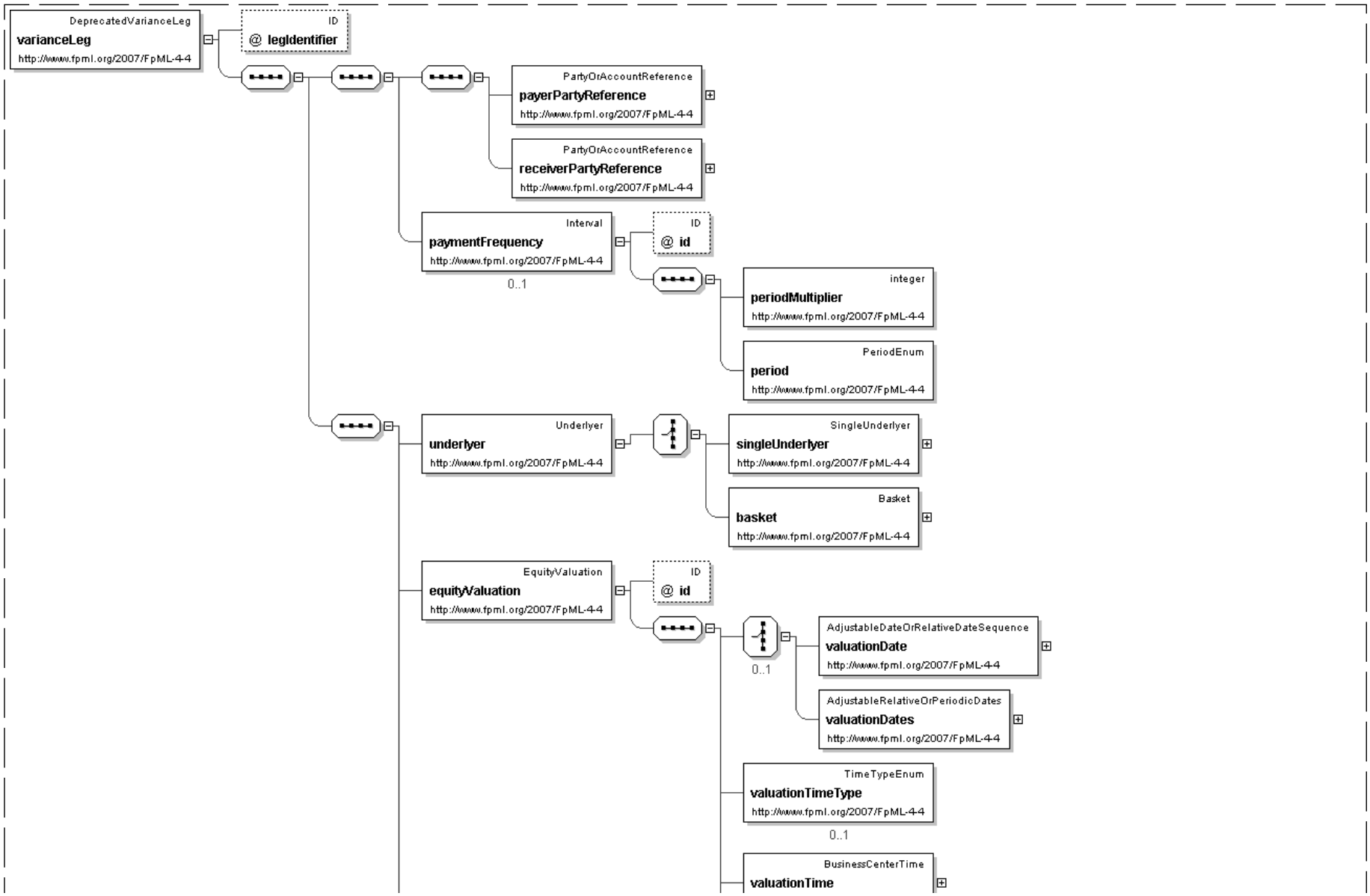
top

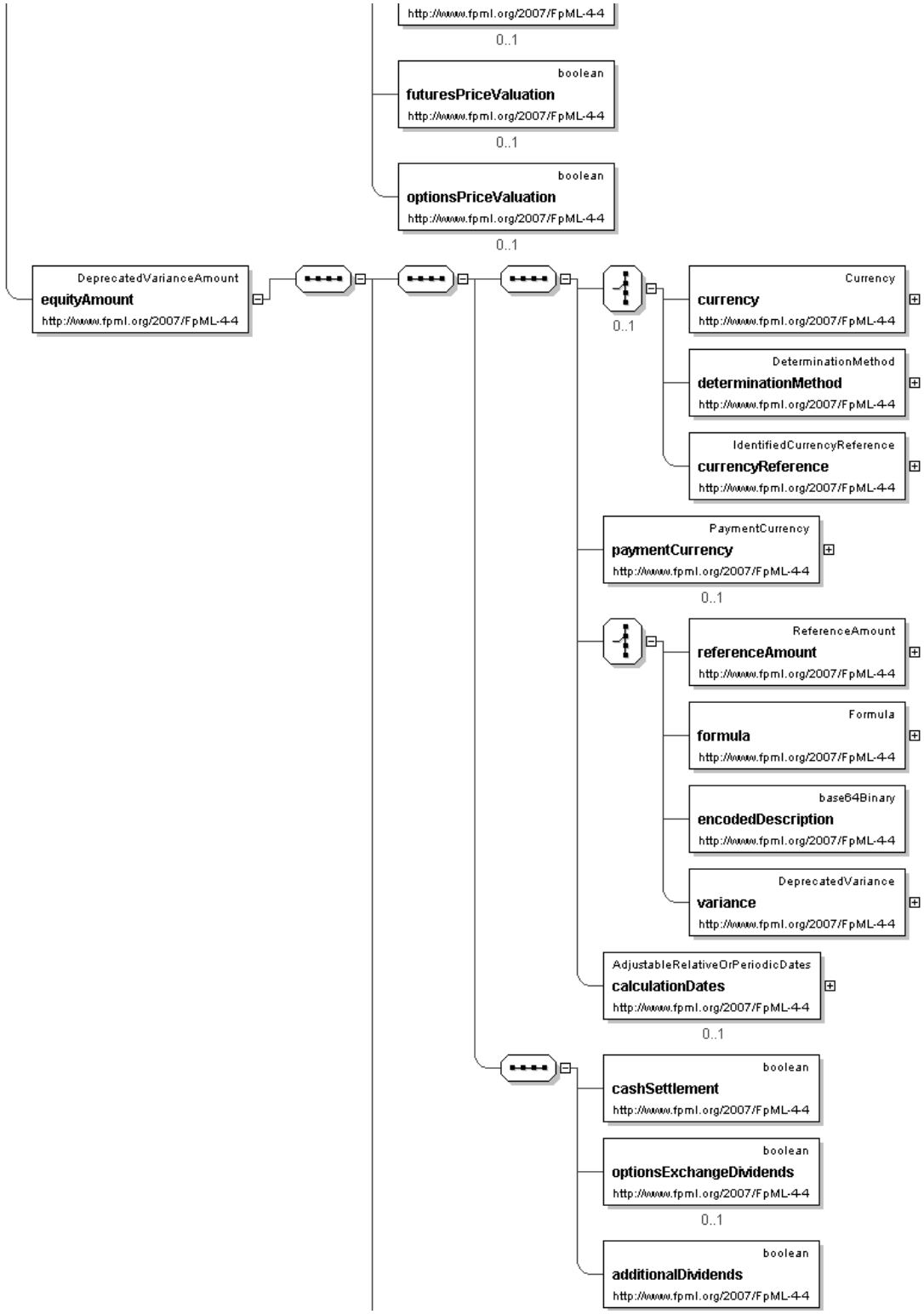
Element: varianceLeg

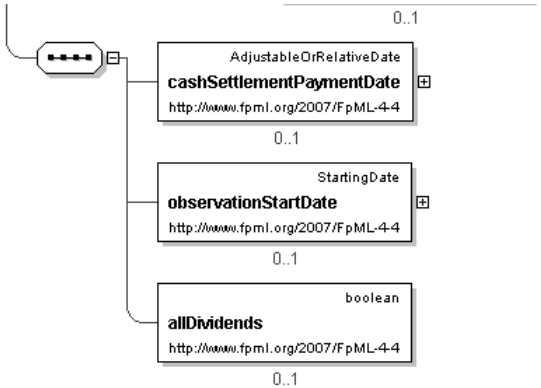
- *This element can be used wherever the following element is referenced:*
  - returnSwapLeg

Name	varianceLeg
Type	DeprecatedVarianceLeg
Nillable	no
Abstract	no
Documentation	DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. The variance leg of the return swap.

Logical Diagram



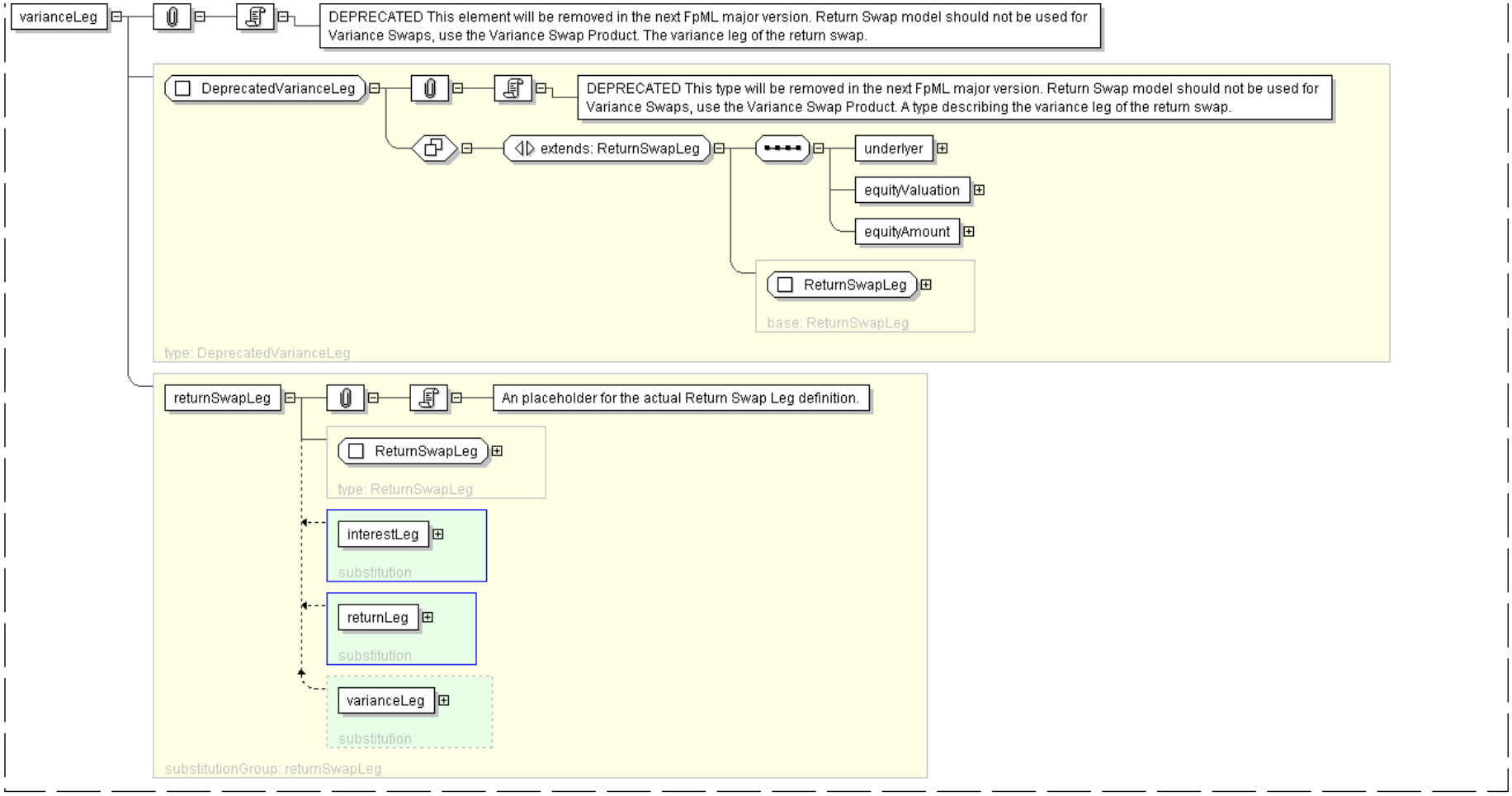

http://www.fpml.org/2007/FpML-4-4
0..1
boolean
futuresPriceValuation
http://www.fpml.org/2007/FpML-4-4
0..1
boolean
optionsPriceValuation
http://www.fpml.org/2007/FpML-4-4
0..1
DeprecatedVarianceAmount
equityAmount
http://www.fpml.org/2007/FpML-4-4
0..1
Currency
currency
http://www.fpml.org/2007/FpML-4-4
DeterminationMethod
determinationMethod
http://www.fpml.org/2007/FpML-4-4
IdentifiedCurrencyReference
currencyReference
http://www.fpml.org/2007/FpML-4-4
PaymentCurrency
paymentCurrency
http://www.fpml.org/2007/FpML-4-4
0..1
ReferenceAmount
referenceAmount
http://www.fpml.org/2007/FpML-4-4
Formula
formula
http://www.fpml.org/2007/FpML-4-4
base64Binary
encodedDescription
http://www.fpml.org/2007/FpML-4-4
DeprecatedVariance
variance
http://www.fpml.org/2007/FpML-4-4
AdjustableRelativeOrPeriodicDates
calculationDates
http://www.fpml.org/2007/FpML-4-4
0..1
boolean
cashSettlement
http://www.fpml.org/2007/FpML-4-4
boolean
optionsExchangeDividends
http://www.fpml.org/2007/FpML-4-4
0..1
boolean
additionalDividends
http://www.fpml.org/2007/FpML-4-4



XML Instance Representation

```
<varianceLeg
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <equityValuation> EquityValuation </equityValuation> [1]
   'Valuation of the underlyer.'

   <equityAmount> DeprecatedVarianceAmount </equityAmount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity
   Payment Date relates. Unless otherwise specified, this term has the meaning defined in the
   ISDA 2002 Equity Derivatives Definitions.'

</varianceLeg>
```

Diagram



Schema Component Representation

```
<xsd:element name="varianceLeg" type=" DeprecatedVarianceLeg
" substitutionGroup="returnSwapLeg" deprecated="true" deprecatedReason="Return Swap
model should not be used for Variance Swaps, use the Variance Swap Product"/>
```

top

Global Definitions

Complex Type: AdditionalDisruptionEvents

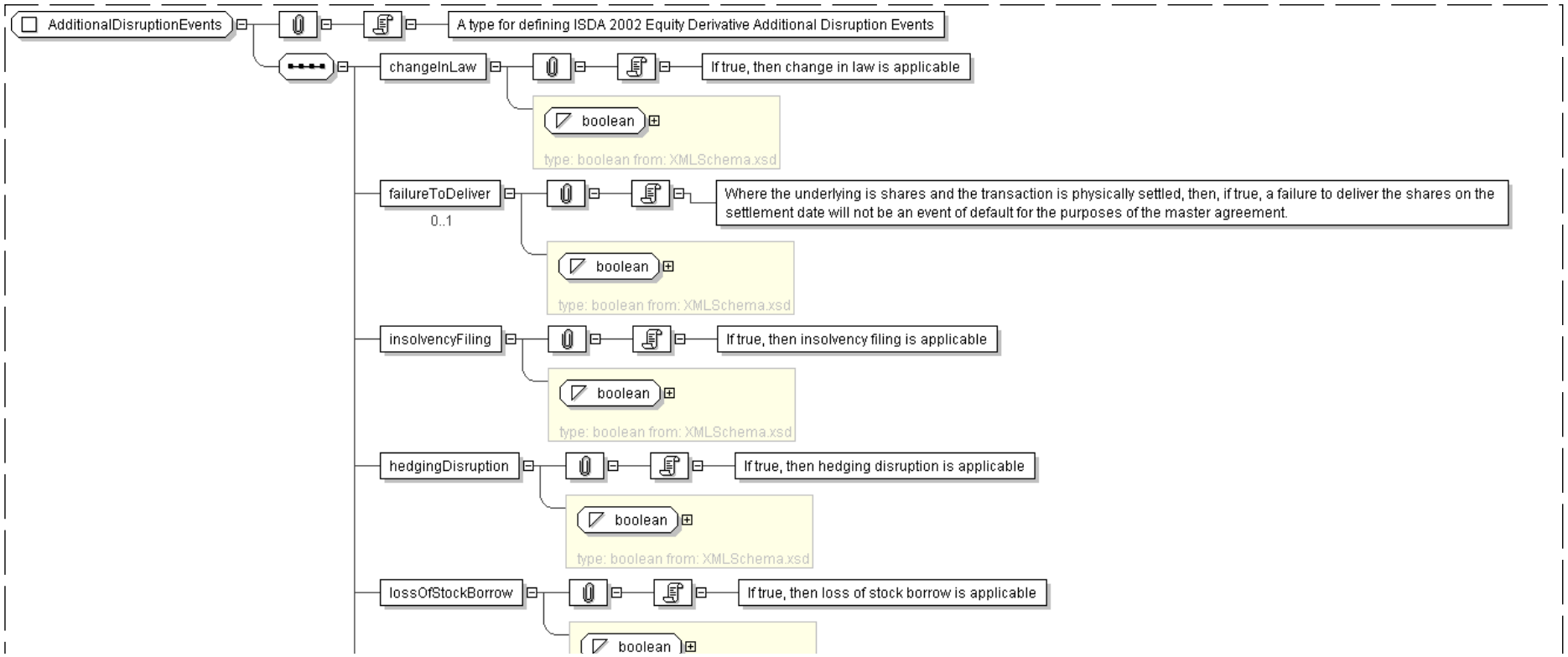
Super-types:	None
Sub-types:	None

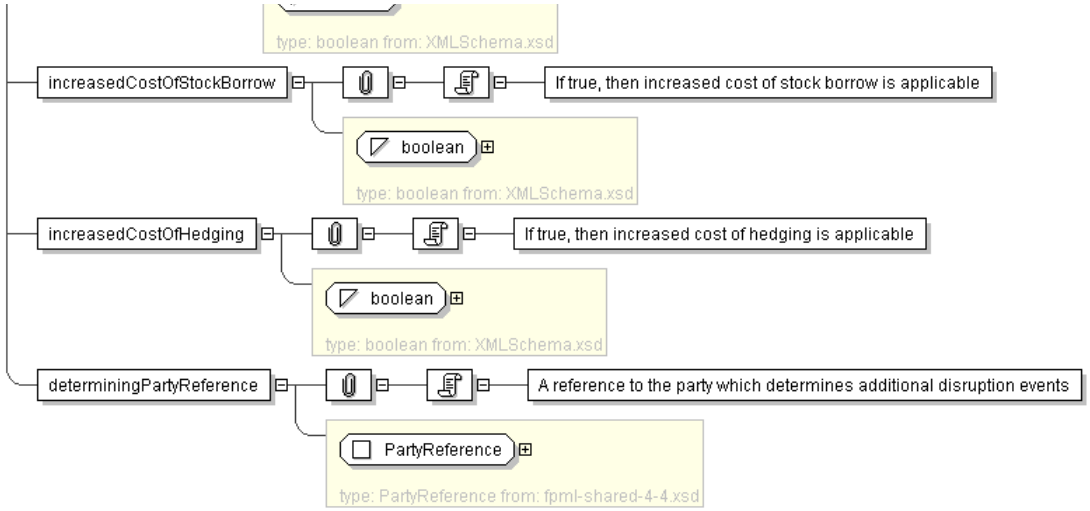
Name	AdditionalDisruptionEvents
Used by (from the same schema document)	Complex Type **ExtraordinaryEvents**
Abstract	no
Documentation	A type for defining ISDA 2002 Equity Derivative Additional Disruption Events

XML Instance Representation

```
<...>
    <changeInLaw> xsd:boolean </changeInLaw> [1]
    'If true, then change in law is applicable'

    <failureToDeliver> xsd:boolean </failureToDeliver> [0..1]
    'Where the underlying is shares and the transaction is physically settled, then, if true,
    a failure to deliver the shares on the settlement date will not be an event of default for
    the purposes of the master agreement.'

    <insolvencyFiling> xsd:boolean </insolvencyFiling> [1]
    'If true, then insolvency filing is applicable'

    <hedgingDisruption> xsd:boolean </hedgingDisruption> [1]
    'If true, then hedging disruption is applicable'

    <lossOfStockBorrow> xsd:boolean </lossOfStockBorrow> [1]
    'If true, then loss of stock borrow is applicable'

    <increasedCostOfStockBorrow> xsd:boolean </increasedCostOfStockBorrow> [1]
    'If true, then increased cost of stock borrow is applicable'

    <increasedCostOfHedging> xsd:boolean </increasedCostOfHedging> [1]
    'If true, then increased cost of hedging is applicable'

    <determiningPartyReference> PartyReference </determiningPartyReference> [1]
    'A reference to the party which determines additional disruption events'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="AdditionalDisruptionEvents">
   <xsd:sequence>
      <xsd:element name="changeInLaw" type=" xsd:boolean "/>
      <xsd:element name="failureToDeliver" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="insolvencyFiling" type=" xsd:boolean "/>
      <xsd:element name="hedgingDisruption" type=" xsd:boolean "/>
      <xsd:element name="lossOfStockBorrow" type=" xsd:boolean "/>
      <xsd:element name="increasedCostOfStockBorrow" type=" xsd:boolean "/>
      <xsd:element name="increasedCostOfHedging" type=" xsd:boolean "/>
      <xsd:element name="determiningPartyReference" type=" PartyReference "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: AdditionalPaymentAmount

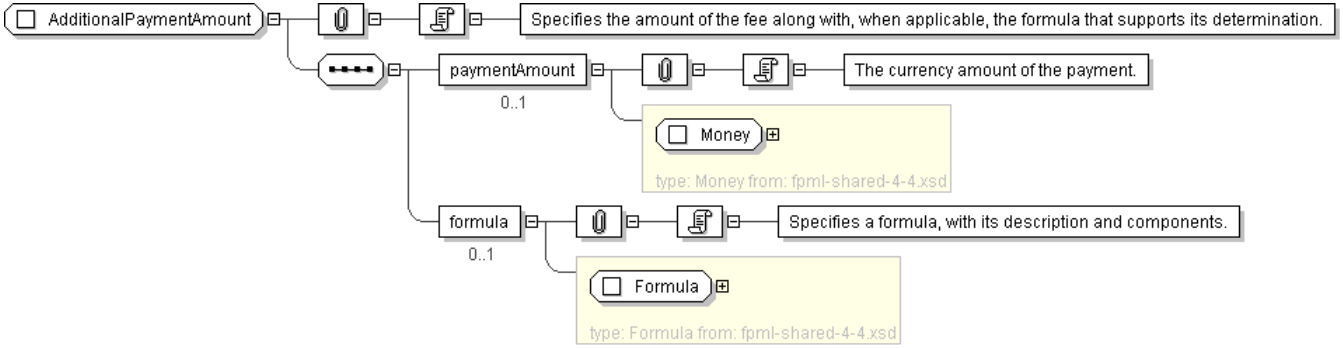
Super-types:	None
Sub-types:	None

Name	AdditionalPaymentAmount
Used by (from the same schema document)	Complex Type **ReturnSwapAdditionalPayment**
Abstract	no
Documentation	Specifies the amount of the fee along with, when applicable, the formula that supports its determination.

XML Instance Representation

```
<...>
   <paymentAmount> Money </paymentAmount> [0..1]
   'The currency amount of the payment.'

   <formula> Formula </formula> [0..1]
   'Specifies a formula, with its description and components.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdditionalPaymentAmount">
   <xsd:sequence>
      <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
      <xsd:element name="formula" type=" Formula " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: AdjustableDateOrRelativeDateSequence

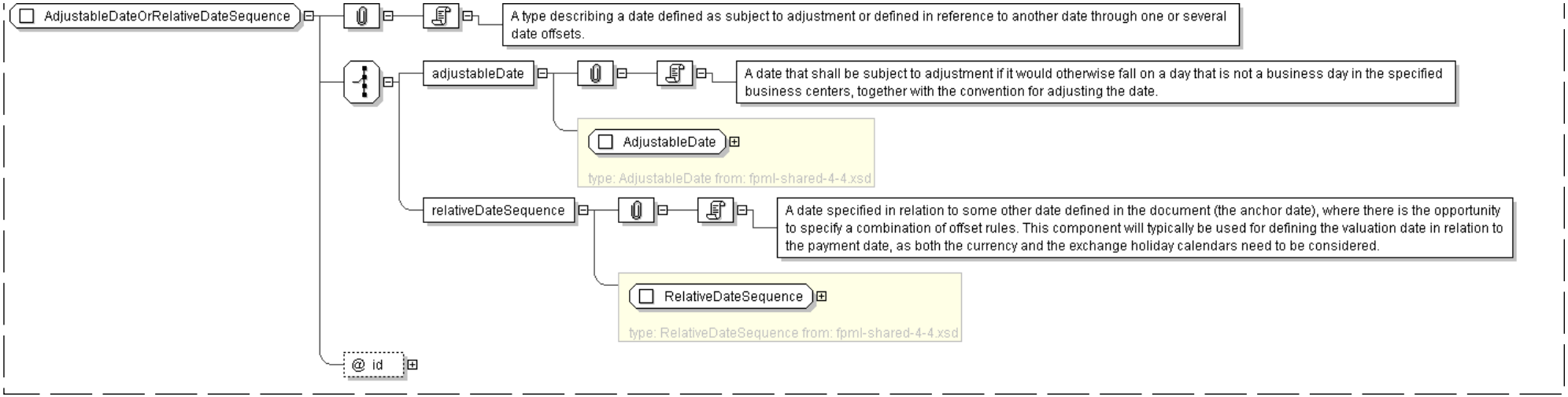
Super-types:	None
Sub-types:	None

Name	AdjustableDateOrRelativeDateSequence
Used by (from the same schema document)	Complex Type **EquityValuation**
Abstract	no
Documentation	A type describing a date defined as subject to adjustment or defined in reference to another date through one or several date offsets.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
Start Choice [1]
      <adjustableDate> AdjustableDate </adjustableDate> [1]
      'A date that shall be subject to adjustment if it would otherwise fall on a day that is not
      a business day in the specified business centers, together with the convention for
      adjusting the date.'

      <relativeDateSequence> RelativeDateSequence </relativeDateSequence> [1]
      'A date specified in relation to some other date defined in the document (the anchor
      date), where there is the opportunity to specify a combination of offset rules. This
      component will typically be used for defining the valuation date in relation to the
      payment date, as both the currency and the exchange holiday calendars need to be considered.'

End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="AdjustableDateOrRelativeDateSequence">
   <xsd:choice>
      <xsd:element name="adjustableDate" type=" AdjustableDate "/>
      <xsd:element name="relativeDateSequence" type=" RelativeDateSequence "/>
   </xsd:choice>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

top

Complex Type: BoundedCorrelation

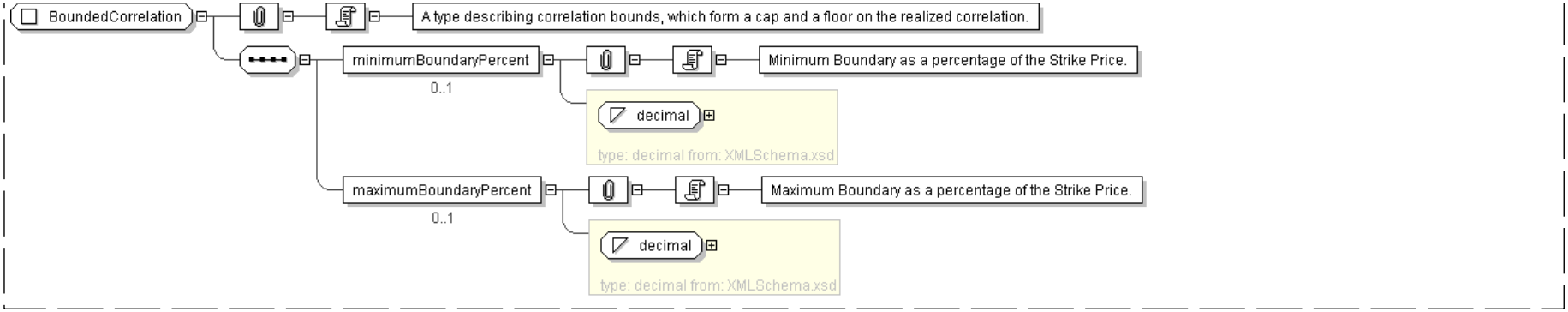
Super-types:	None
Sub-types:	None

Name	BoundedCorrelation
Used by (from the same schema document)	Complex Type **Correlation**
Abstract	no
Documentation	A type describing correlation bounds, which form a cap and a floor on the realized correlation.

XML Instance Representation

```
<...>
   <minimumBoundaryPercent> xsd:decimal </minimumBoundaryPercent> [0..1]
   'Minimum Boundary as a percentage of the Strike Price.'

   <maximumBoundaryPercent> xsd:decimal </maximumBoundaryPercent> [0..1]
   'Maximum Boundary as a percentage of the Strike Price.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BoundedCorrelation">
   <xsd:sequence>
      <xsd:element name="minimumBoundaryPercent" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="maximumBoundaryPercent" type=" xsd:decimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: BoundedVariance

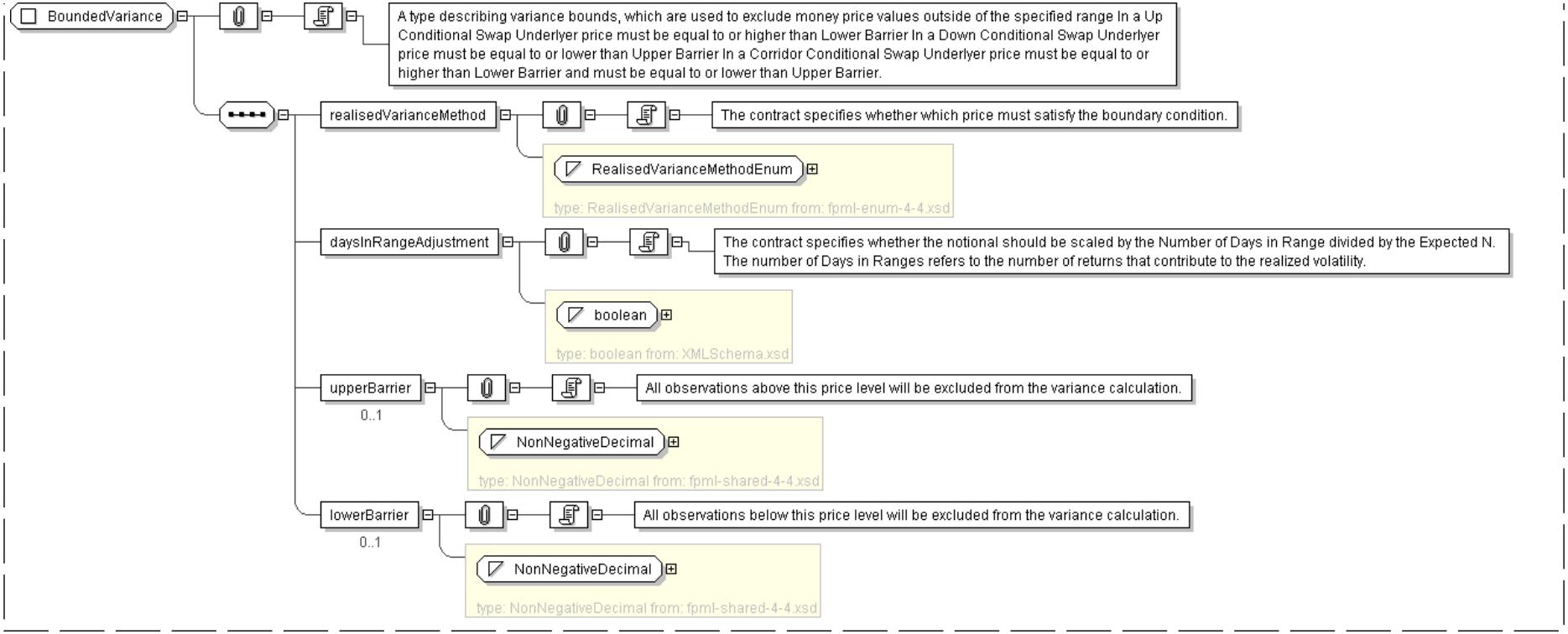
Super-types:	None
Sub-types:	None

Name	BoundedVariance
Used by (from the same schema document)	Complex Type **Variance**
Abstract	no
Documentation	A type describing variance bounds, which are used to exclude money price values outside of the specified range In a Up Conditional Swap Underlyer price must be equal to or higher than Lower Barrier In a Down Conditional Swap Underlyer price must be equal to or lower than Upper Barrier In a Corridor Conditional Swap Underlyer price must be equal to or higher than Lower Barrier and must be equal to or lower than Upper Barrier.

XML Instance Representation

```
<...>
   <realisedVarianceMethod> RealisedVarianceMethodEnum </realisedVarianceMethod> [1]
   'The contract specifies whether which price must satisfy the boundary condition.'

   <daysInRangeAdjustment> xsd:boolean </daysInRangeAdjustment> [1]
   'The contract specifies whether the notional should be scaled by the Number of Days in
   Range divided by the Expected N. The number of Days in Ranges refers to the number of
   returns that contribute to the realized volatility.'

   <upperBarrier> NonNegativeDecimal </upperBarrier> [0..1]
   'All observations above this price level will be excluded from the variance calculation.'

   <lowerBarrier> NonNegativeDecimal </lowerBarrier> [0..1]
   'All observations below this price level will be excluded from the variance calculation.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="BoundedVariance">
   <xsd:sequence>
      <xsd:element name="realisedVarianceMethod" type=" RealisedVarianceMethodEnum "/>
      <xsd:element name="daysInRangeAdjustment" type=" xsd:boolean "/>
      <xsd:element name="upperBarrier" type=" NonNegativeDecimal " minOccurs="0"/>
      <xsd:element name="lowerBarrier" type=" NonNegativeDecimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: CalculatedAmount

Super-types:	None
Sub-types:	None

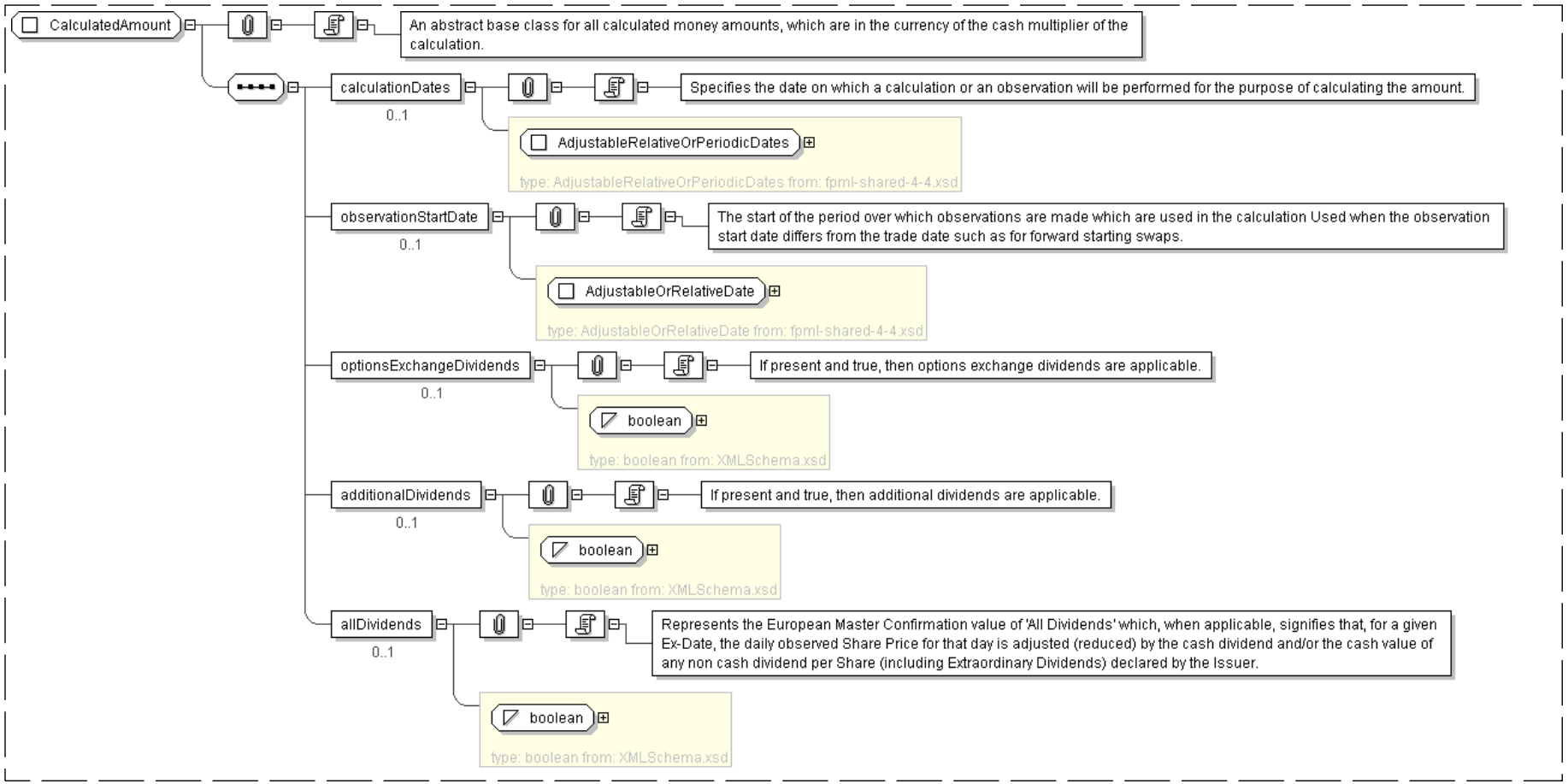
Name	CalculatedAmount
Abstract	yes
Documentation	An abstract base class for all calculated money amounts, which are in the currency of the cash multiplier of the calculation.

XML Instance Representation

```
<...>
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for the
   purpose of calculating the amount.'

   <observationStartDate> AdjustableOrRelativeDate </observationStartDate> [0..1]
   'The start of the period over which observations are made which are used in the
```

```
    calculation Used when the observation start date differs from the trade date such as
    for forward starting swaps.'

    <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
    'If present and true, then options exchange dividends are applicable.'

    <additionalDividends> xsd:boolean </additionalDividends> [0..1]
    'If present and true, then additional dividends are applicable.'

    <allDividends> xsd:boolean </allDividends> [0..1]
    'Represents the European Master Confirmation value of \'All Dividends\' which, when
    applicable, signifies that, for a given Ex-Date, the daily observed Share Price for that day
    is adjusted (reduced) by the cash dividend and/or the cash value of any non cash dividend
    per Share (including Extraordinary Dividends) declared by the Issuer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculatedAmount" abstract="true">
   <xsd:sequence>
      <xsd:element name="calculationDates" type=" AdjustableRelativeOrPeriodicDates " minOccurs="0"/>
      <xsd:element name="observationStartDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
      <xsd:element name="optionsExchangeDividends" type=" xsd:boolean " minOccurs="0"/>
```

```
      <xsd:element name="additionalDividends" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="allDividends" type=" xsd:boolean " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: CalculationFromObservation

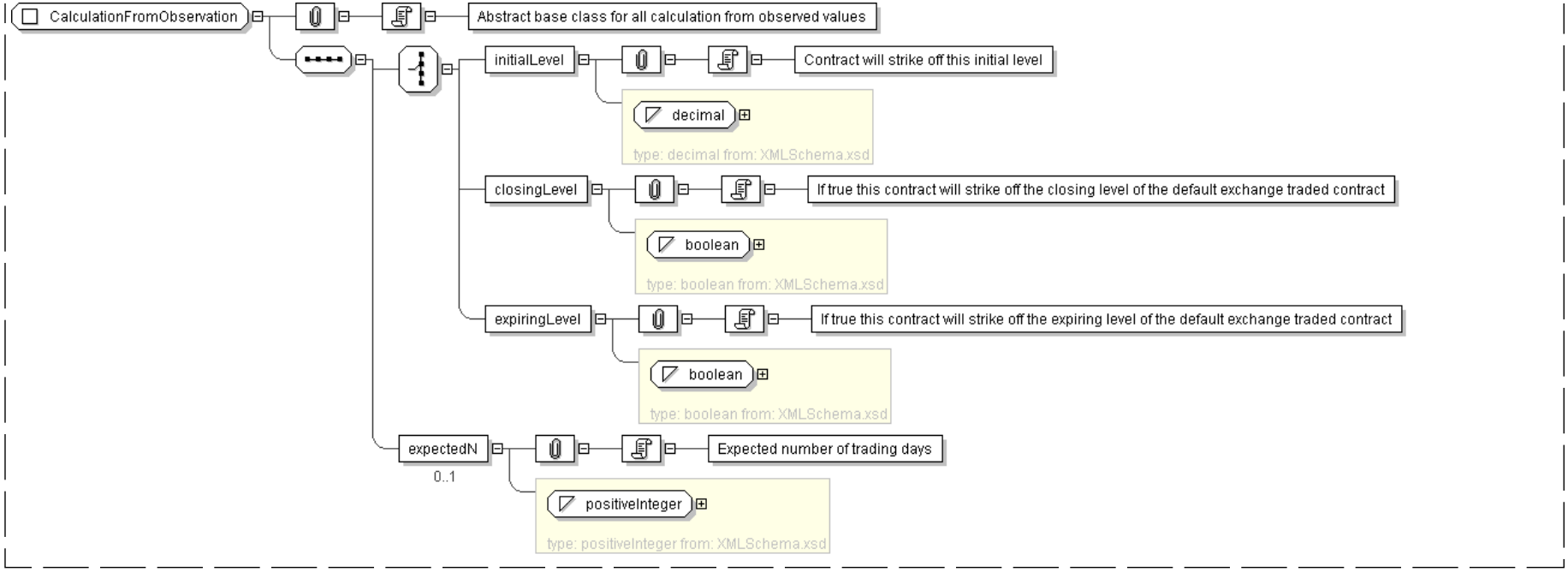
Super-types:	None
Sub-types:	• Correlation (by extension) • Variance (by extension)

Name	CalculationFromObservation
Abstract	yes
Documentation	Abstract base class for all calculation from observed values

XML Instance Representation

```
<...>
Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      'Contract will strike off this initial level'

      <closingLevel> xsd:boolean </closingLevel> [1]
      'If true this contract will strike off the closing level of the default exchange
      traded contract'

      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If true this contract will strike off the expiring level of the default exchange
      traded contract'

End Choice
   <expectedN> xsd:positiveInteger </expectedN> [0..1]
   'Expected number of trading days'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CalculationFromObservation" abstract="true">
   <xsd:sequence>
      <xsd:choice>
         <xsd:element name="initialLevel" type=" xsd:decimal "/>
         <xsd:element name="closingLevel" type=" xsd:boolean "/>
         <xsd:element name="expiringLevel" type=" xsd:boolean "/>
      </xsd:choice>
      <xsd:element name="expectedN" type=" xsd:positiveInteger " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: Compounding

Super-types:	None
Sub-types:	None

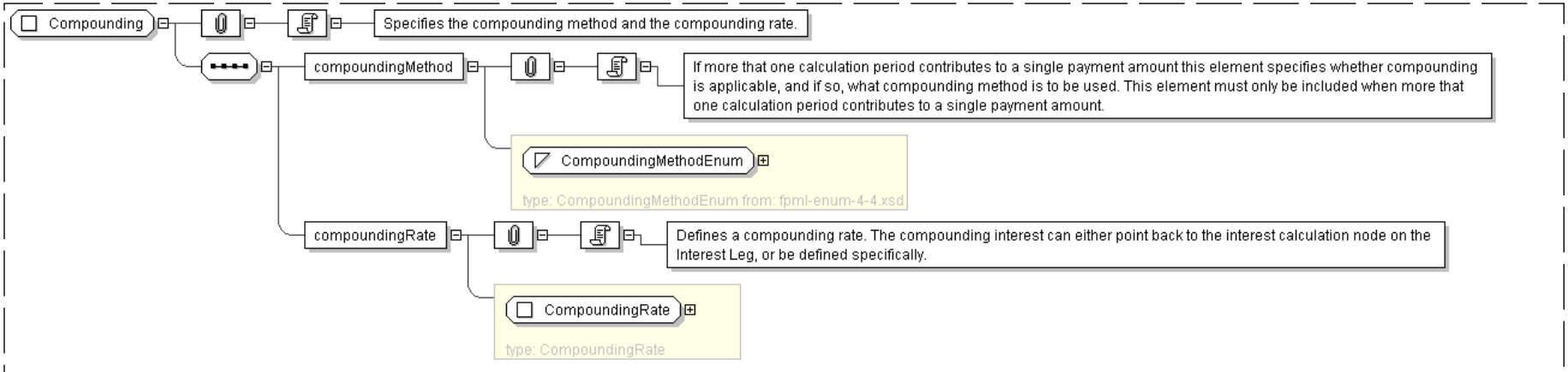
Name	Compounding
Used by (from the same schema document)	Complex Type **InterestCalculation**
Abstract	no
Documentation	Specifies the compounding method and the compounding rate.

XML Instance Representation

```
<...>
   <compoundingMethod> CompoundingMethodEnum </compoundingMethod> [1]
   'If more that one calculation period contributes to a single payment amount this
   element specifies whether compounding is applicable, and if so, what compounding method is
   to be used. This element must only be included when more that one calculation
   period contributes to a single payment amount.'

   <compoundingRate> CompoundingRate </compoundingRate> [1]
```

```
    'Defines a compounding rate. The compounding interest can either point back to the
    interest calculation node on the Interest Leg, or be defined specifically.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Compounding">
   <xsd:sequence>
      <xsd:element name="compoundingMethod" type=" CompoundingMethodEnum "/>
      <xsd:element name="compoundingRate" type=" CompoundingRate "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: CompoundingRate

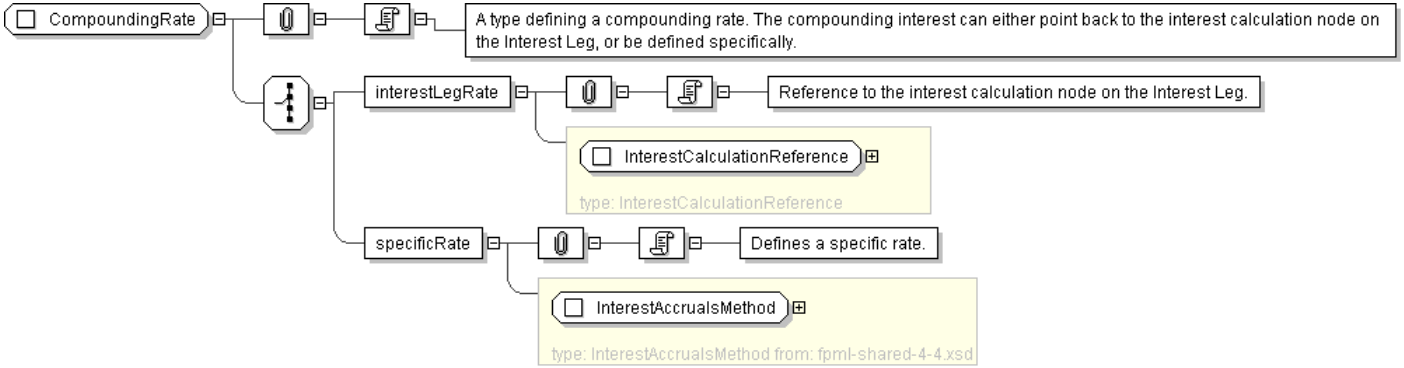
Super-types:	None
Sub-types:	None

Name	CompoundingRate
Used by (from the same schema document)	Complex Type **Compounding**
Abstract	no
Documentation	A type defining a compounding rate. The compounding interest can either point back to the interest calculation node on the Interest Leg, or be defined specifically.

XML Instance Representation

```
<...>
Start Choice [1]
      <interestLegRate> InterestCalculationReference </interestLegRate> [1]
      'Reference to the interest calculation node on the Interest Leg.'

      <specificRate> InterestAccrualsMethod </specificRate> [1]
      'Defines a specific rate.'

End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="CompoundingRate">
   <xsd:choice>
      <xsd:element name="interestLegRate" type=" InterestCalculationReference "/>
      <xsd:element name="specificRate" type=" InterestAccrualsMethod "/>
   </xsd:choice>
</xsd:complexType>
```

top

Complex Type: Correlation

Super-types:	CalculationFromObservation < **Correlation** (by extension)
Sub-types:	None

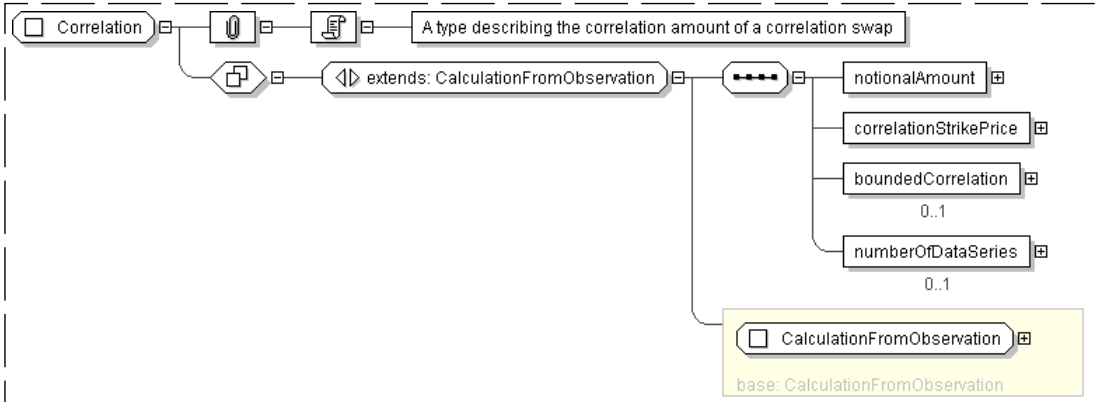
Name	Correlation
Abstract	no
Documentation	A type describing the correlation amount of a correlation swap

XML Instance Representation

```
<...>
Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      'Contract will strike off this initial level'

      <closingLevel> xsd:boolean </closingLevel> [1]
      'If true this contract will strike off the closing level of the default exchange
      traded contract'

      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If true this contract will strike off the expiring level of the default exchange
      traded contract'

End Choice
   <expectedN> xsd:positiveInteger </expectedN> [0..1]
   'Expected number of trading days'

   <notionalAmount> Money </notionalAmount> [1]
   'Notional amount, which is a cash multiplier'

   <correlationStrikePrice> CorrelationValue </correlationStrikePrice> [1]
   'Correlation Strike Price'

   <boundedCorrelation> BoundedCorrelation </boundedCorrelation> [0..1]
```

```
    'Bounded Correlation'

    <numberOfDataSeries> xsd:positiveInteger </numberOfDataSeries> [0..1]
    'Number of data series, normal market practice is that correlation data sets are drawn
    from geographic market areas, such as America, Europe and Asia Pacific, each of
    these geographic areas will have its own data series to avoid contagion'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Correlation">
   <xsd:complexContent>
      <xsd:extension base=" CalculationFromObservation ">
         <xsd:sequence>
            <xsd:element name="notionalAmount" type=" Money "/>
            <xsd:element name="correlationStrikePrice" type=" CorrelationValue "/>
            <xsd:element name="boundedCorrelation" type=" BoundedCorrelation " minOccurs="0"/>
            <xsd:element name="numberOfDataSeries" type=" xsd:positiveInteger " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DeprecatedVariance

Super-types:	None
Sub-types:	None

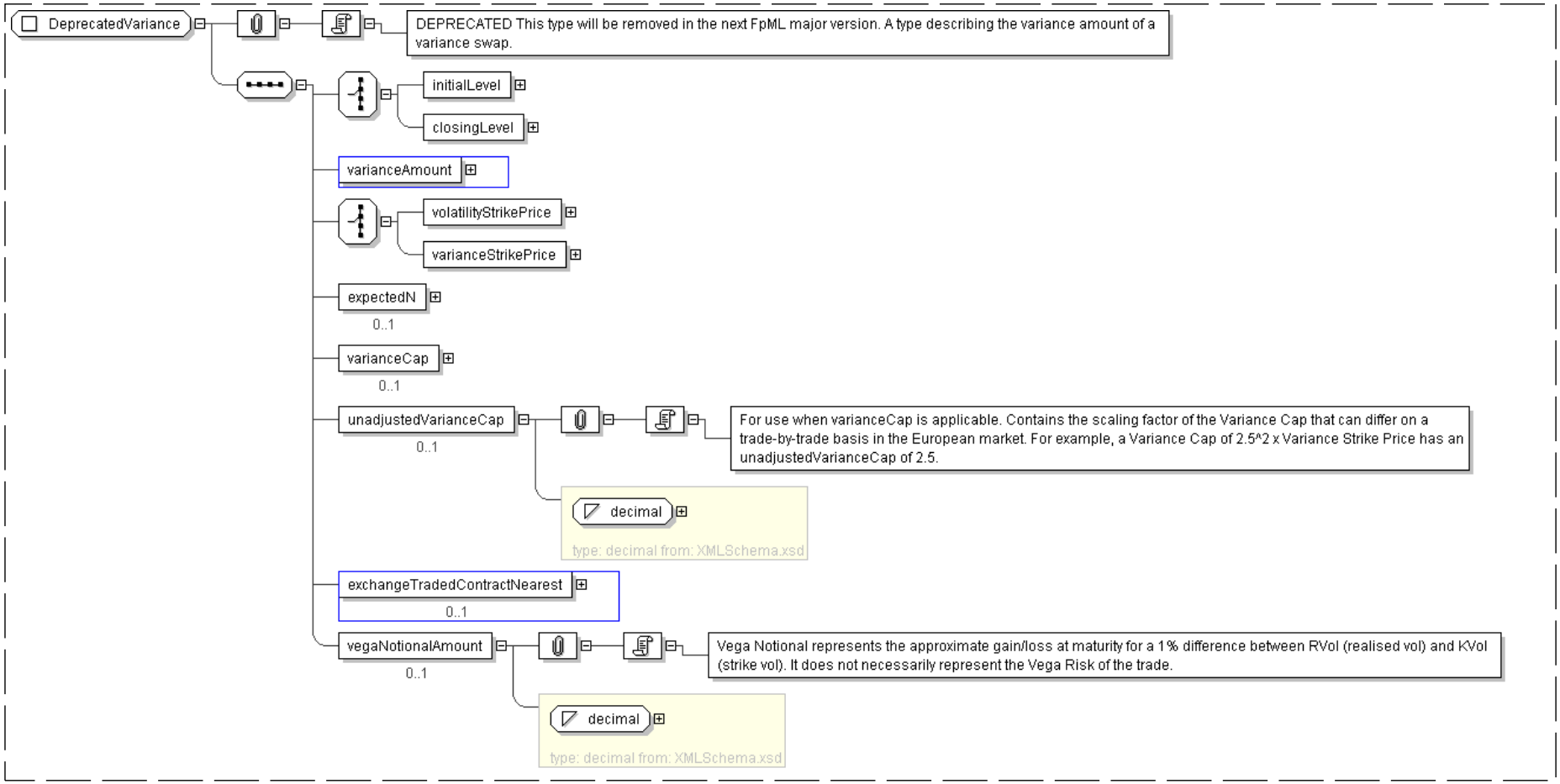
Name	DeprecatedVariance
Used by (from the same schema document)	Complex Type **LegAmount**
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. A type describing the variance amount of a variance swap.

XML Instance Representation

```
<...>
Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      <closingLevel> xsd:boolean </closingLevel> [1]
End Choice
   <varianceAmount> Money </varianceAmount> [1]
```

```
Start Choice [1]
      <volatilityStrikePrice> xsd:decimal </volatilityStrikePrice> [1]
      <varianceStrikePrice> xsd:decimal </varianceStrikePrice> [1]
End Choice
   <expectedN> xsd:integer </expectedN> [0..1]
   <varianceCap> xsd:boolean </varianceCap> [0..1]
   <unadjustedVarianceCap> xsd:decimal </unadjustedVarianceCap> [0..1]
   'For use when varianceCap is applicable. Contains the scaling factor of the Variance Cap
   that can differ on a trade-by-trade basis in the European market. For example, a Variance
   Cap of 2.5^2 x Variance Strike Price has an unadjustedVarianceCap of 2.5.'

   <exchangeTradedContractNearest> ExchangeTradedContract </exchangeTradedContractNearest> [0..1]
   <vegaNotionalAmount> xsd:decimal </vegaNotionalAmount> [0..1]
   'Vega Notional represents the approximate gain/loss at maturity for a 1% difference
   between RVol (realised vol) and KVol (strike vol). It does not necessarily represent the
   Vega Risk of the trade.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVariance" deprecated="true" deprecatedReason="Use new
Variance complex type">
   <xsd:sequence>
```

```
      <xsd:choice>
         <xsd:element name="initialLevel" type=" xsd:decimal "/>
         <xsd:element name="closingLevel" type=" xsd:boolean "/>
      </xsd:choice>
      <xsd:element name="varianceAmount" type=" Money "/>
      <xsd:choice>
         <xsd:element name="volatilityStrikePrice" type=" xsd:decimal "/>
         <xsd:element name="varianceStrikePrice" type=" xsd:decimal "/>
      </xsd:choice>
      <xsd:element name="expectedN" type=" xsd:integer " minOccurs="0"/>
      <xsd:element name="varianceCap" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="unadjustedVarianceCap" type=" xsd:decimal " minOccurs="0"/>
      <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract "
      minOccurs="0"/>
      <xsd:element name="vegaNotionalAmount" type=" xsd:decimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: DeprecatedVarianceAmount

Super-types:	LegAmount < ReturnSwapAmount (by extension) < **DeprecatedVarianceAmount** (by extension)
Sub-types:	None

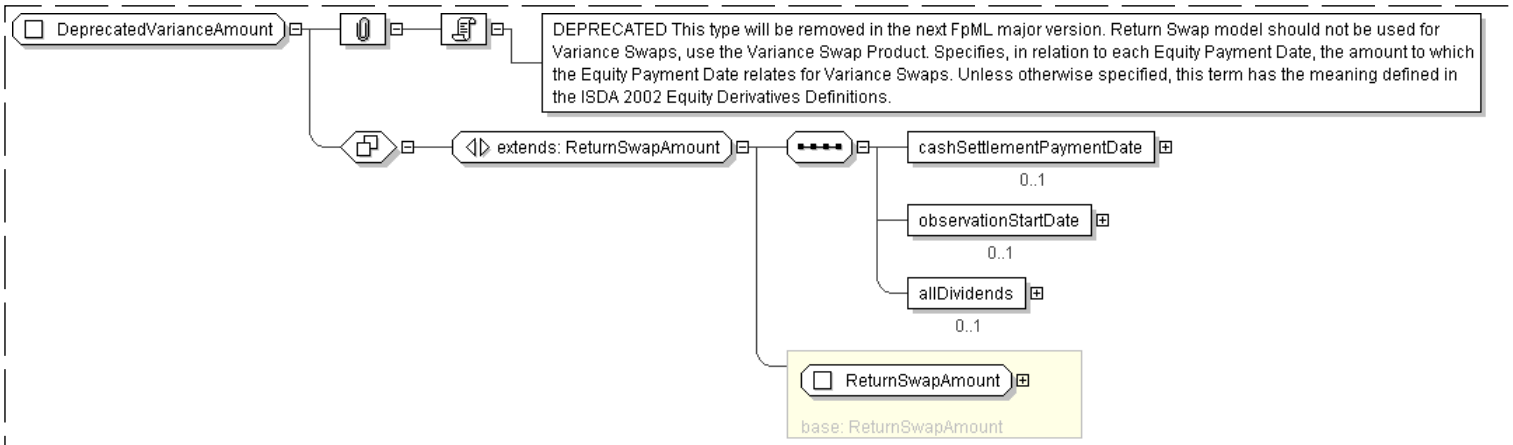
Name	DeprecatedVarianceAmount
Used by (from the same schema document)	Complex Type **DeprecatedVarianceLeg**
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. Specifies, in relation to each Equity Payment Date, the amount to which the Equity Payment Date relates for Variance Swaps. Unless otherwise specified, this term has the meaning defined in the ISDA 2002 Equity Derivatives Definitions.

XML Instance Representation

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

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

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

End Choice
   <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
   'Currency in which the payment relating to the leg amount (equity amount or interest amount)
   or the dividend will be denominated.'

Start Choice [1]
      <referenceAmount> ReferenceAmount </referenceAmount> [1]
      'Specifies the reference Amount when this term either corresponds to the standard
      ISDA Definition (either the 2002 Equity Definition for the Equity Amount, or the
      2000 Definition for the Interest Amount), or points to a term defined elsewhere in the
      swap document.'

      <formula> Formula </formula> [1]
      'Specifies a formula, with its description and components.'

      <encodedDescription> xsd:base64Binary </encodedDescription> [1]
      'Description of the leg amount when represented through an encoded image.'
```

```
      <variance> DeprecatedVariance </variance> [1]
      'DEPRECATED This element will be removed in the next FpML major version. Return Swap
      model should not be used for Variance Swaps, use the Variance Swap Product. Specifies
      Variance for Variance Leg.'

End Choice
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for the
   purpose of defining the Equity Amount, and in accordance to the definition terms of
   this latter.'

   <cashSettlement> xsd:boolean </cashSettlement> [1]
   'If true, then cash settlement is applicable.'

   <optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]
   'If present and true, then options exchange dividends are applicable.'

   <additionalDividends> xsd:boolean </additionalDividends> [0..1]
   'If present and true, then additional dividends are applicable.'

   <cashSettlementPaymentDate> AdjustableOrRelativeDate </cashSettlementPaymentDate> [0..1]
   'Typically specified as a number of days following the valuation date, such as one
   settlement cycle following the valuation date. Number of days can vary in the European market.'

   <observationStartDate> StartingDate </observationStartDate> [0..1]
   'The start of the period over which observations are made to determine the variance. Used
   when the date differs from the trade date such as for forward starting variance swaps.'

   <allDividends> xsd:boolean </allDividends> [0..1]
   'Represents the European Master Confirmation value of \"All Dividends\" which, when
   applicable, signifies that, for a given Ex-Date, the daily observed Share Price for that day
   is adjusted (reduced) by the cash dividend and/or the cash value of any non cash dividend
   per Share (including Extraordinary Dividends) declared by the Issuer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVarianceAmount" deprecated="true"
deprecatedReason="Return Swap model should not be used for Variance Swaps, use the
Variance Swap Product">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapAmount ">
```

```
        <xsd:sequence>
           <xsd:element name="cashSettlementPaymentDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
           <xsd:element name="observationStartDate" type=" StartingDate " minOccurs="0"/>
           <xsd:element name="allDividends" type=" xsd:boolean " minOccurs="0"/>
        </xsd:sequence>
     </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DeprecatedVarianceLeg

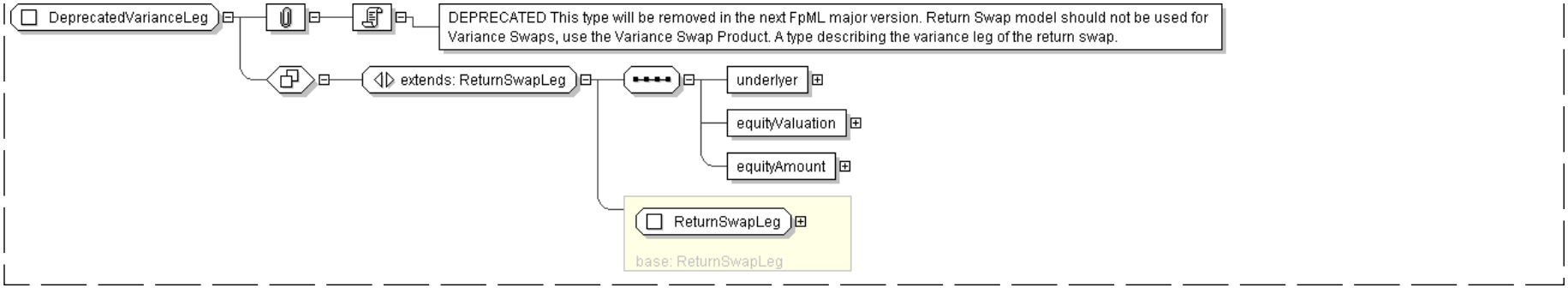
Super-types:	Leg < ReturnSwapLeg (by extension) < **DeprecatedVarianceLeg** (by extension)
Sub-types:	None

Name	DeprecatedVarianceLeg
Used by (from the same schema document)	Element **varianceLeg**
Abstract	no
Documentation	DEPRECATED This type will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. A type describing the variance leg of the return swap.

XML Instance Representation

```
<...
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

   <underlyer> Underlyer </underlyer> [1]
   'Specifies the underlyer of the leg.'

   <equityValuation> EquityValuation </equityValuation> [1]
   'Valuation of the underlyer.'

   <equityAmount> DeprecatedVarianceAmount </equityAmount> [1]
   'Specifies, in relation to each Equity Payment Date, the amount to which the Equity
   Payment Date relates. Unless otherwise specified, this term has the meaning defined in the
   ISDA 2002 Equity Derivatives Definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DeprecatedVarianceLeg" deprecated="true" deprecatedReason="Return
Swap model should not be used for Variance Swaps, use the Variance Swap Product">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="underlyer" type=" Underlyer "/>
            <xsd:element name="equityValuation" type=" EquityValuation "/>
            <xsd:element name="equityAmount" type=" DeprecatedVarianceAmount "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DirectionalLeg

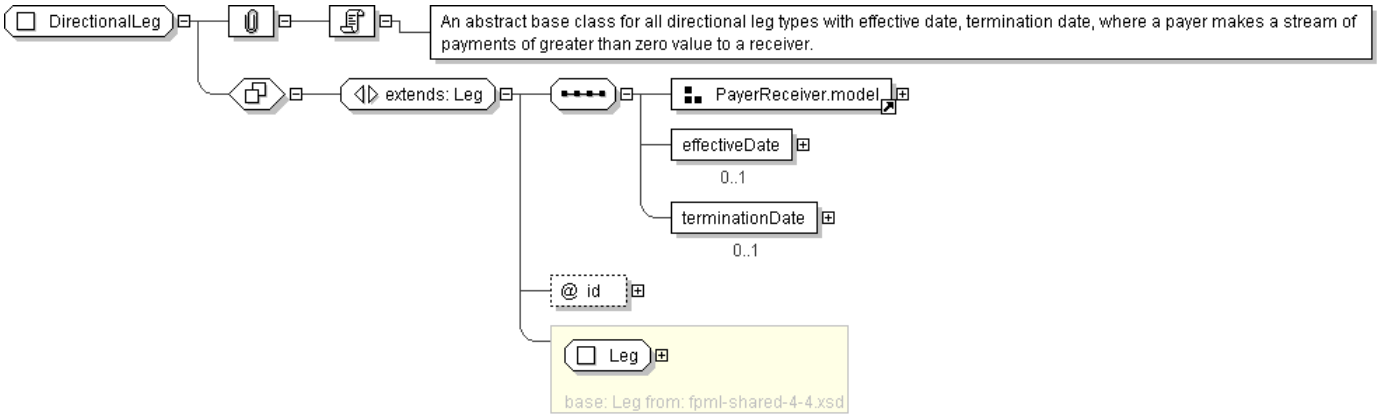
Super-types:	Leg < **DirectionalLeg** (by extension)
Sub-types:	• DirectionalLegUnderlyer (by extension) ○ DirectionalLegUnderlyerValuation (by extension)

Name	DirectionalLeg
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [0..1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component),
   this element will typically point to the effective date of the other leg of the swap.'

   <terminationDate> AdjustableOrRelativeDate </terminationDate> [0..1]
   'Specifies the termination date of this leg of the swap. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component),
   this element will typically point to the termination date of the other leg of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLeg" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Leg ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
            <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
         </xsd:sequence>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DirectionalLegUnderlyer

Super-types:	Leg < DirectionalLeg (by extension) < **DirectionalLegUnderlyer** (by extension)
Sub-types:	• DirectionalLegUnderlyerValuation (by extension)

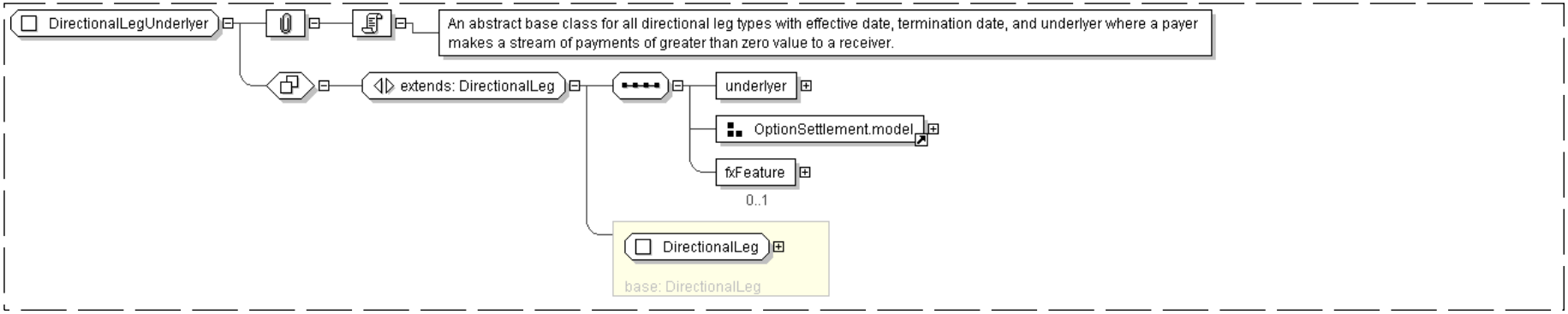
Name	DirectionalLegUnderlyer
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, and underlyer where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

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

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLegUnderlyer" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" DirectionalLeg ">
         <xsd:sequence>
            <xsd:element name="underlyer" type=" Underlyer "/>
            <xsd:group ref=" OptionSettlement.model "/>
            <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DirectionalLegUnderlyerValuation

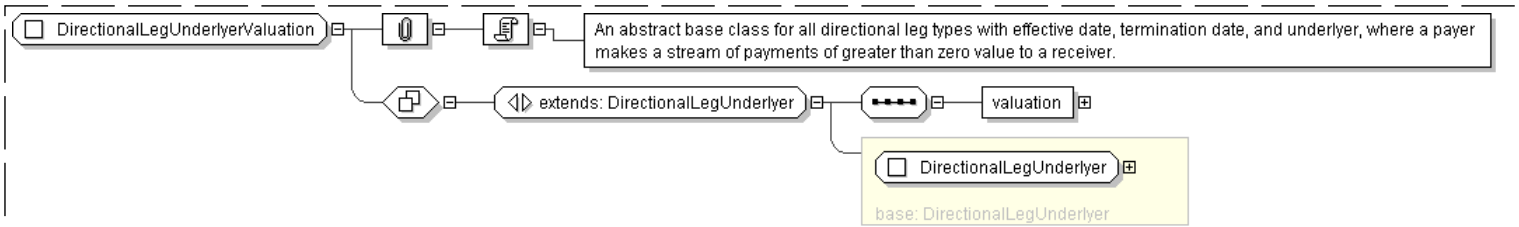
Super-types:	Leg < DirectionalLeg (by extension) < DirectionalLegUnderlyer (by extension) < **DirectionalLegUnderlyerValuation** (by extension)
Sub-types:	None

Name	DirectionalLegUnderlyerValuation
Abstract	yes
Documentation	An abstract base class for all directional leg types with effective date, termination date, and underlyer, where a payer makes a stream of payments of greater than zero value to a receiver.

XML Instance Representation

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

   <valuation> EquityValuation </valuation> [1]
   'Valuation of the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DirectionalLegUnderlyerValuation" abstract="true">
```

```
   <xsd:complexContent>
      <xsd:extension base=" DirectionalLegUnderlyer ">
         <xsd:sequence>
            <xsd:element name="valuation" type=" EquityValuation "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: DividendAdjustment

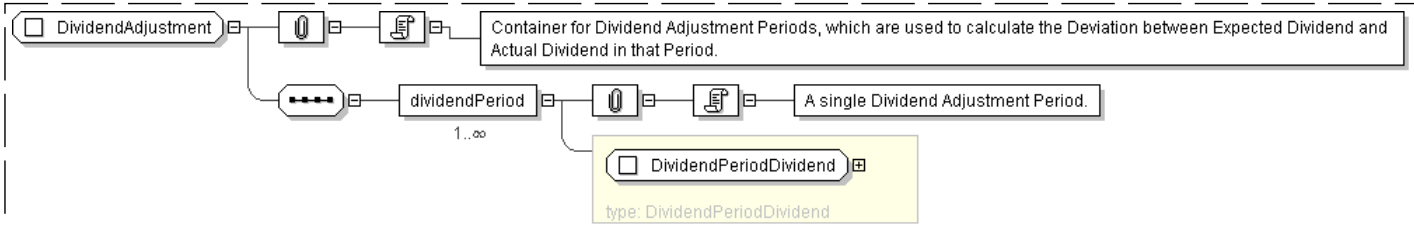
Super-types:	None
Sub-types:	None

Name	DividendAdjustment
Used by (from the same schema document)	Complex Type **OptionFeatures**
Abstract	no
Documentation	Container for Dividend Adjustment Periods, which are used to calculate the Deviation between Expected Dividend and Actual Dividend in that Period.

XML Instance Representation

```
<...>
   <dividendPeriod> DividendPeriodDividend </dividendPeriod> [1..*]
   'A single Dividend Adjustment Period.'
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendAdjustment">
   <xsd:sequence>
      <xsd:element name="dividendPeriod" type=" DividendPeriodDividend " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: DividendPeriod

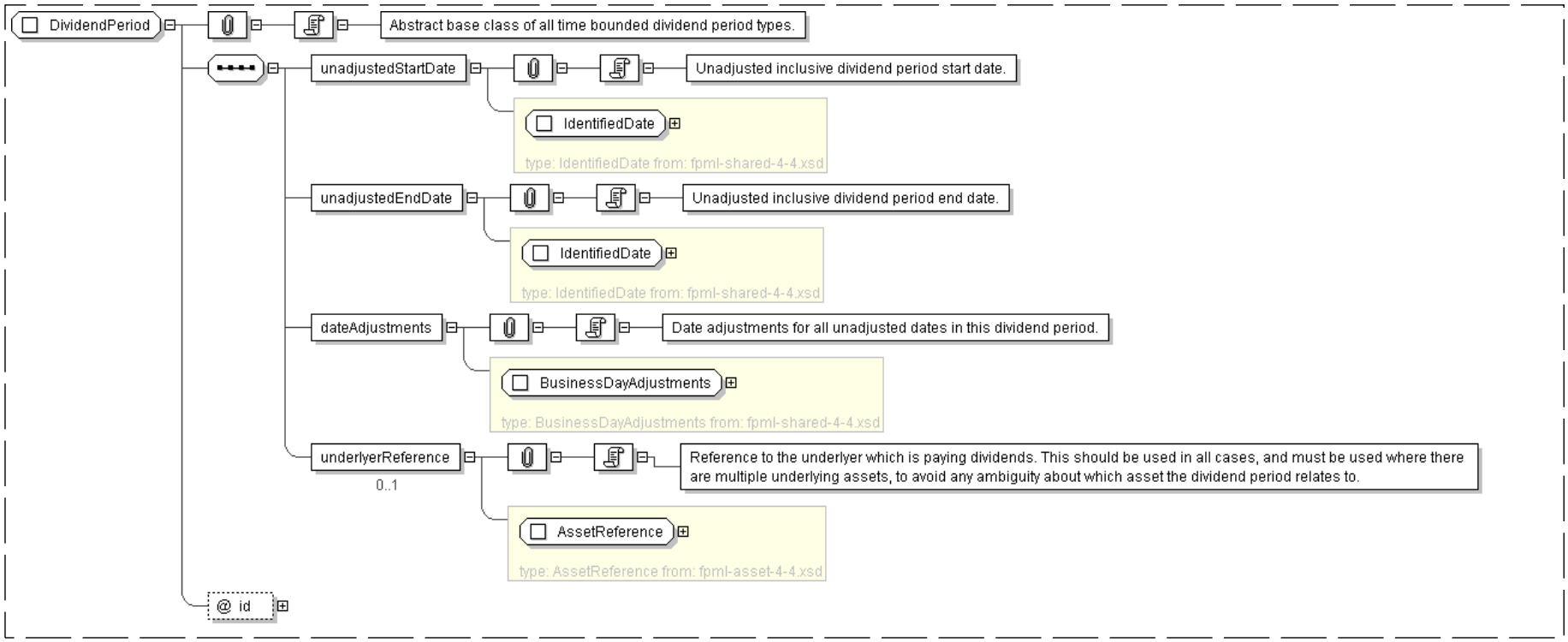
Super-types:	None
Sub-types:	• DividendPeriodDividend (by extension)

Name	DividendPeriod
Abstract	yes
Documentation	Abstract base class of all time bounded dividend period types.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <unadjustedStartDate> IdentifiedDate </unadjustedStartDate> [1]
   'Unadjusted inclusive dividend period start date.'

   <unadjustedEndDate> IdentifiedDate </unadjustedEndDate> [1]
   'Unadjusted inclusive dividend period end date.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
   'Date adjustments for all unadjusted dates in this dividend period.'

   <underlyerReference> AssetReference </underlyerReference> [0..1]
   'Reference to the underlyer which is paying dividends. This should be used in all cases,
   and must be used where there are multiple underlying assets, to avoid any ambiguity about
   which asset the dividend period relates to.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPeriod" abstract="true">
   <xsd:sequence>
      <xsd:element name="unadjustedStartDate" type=" IdentifiedDate "/>
      <xsd:element name="unadjustedEndDate" type=" IdentifiedDate "/>
      <xsd:element name="dateAdjustments" type=" BusinessDayAdjustments "/>
      <xsd:element name="underlyerReference" type=" AssetReference " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

top

Complex Type: DividendPeriodDividend

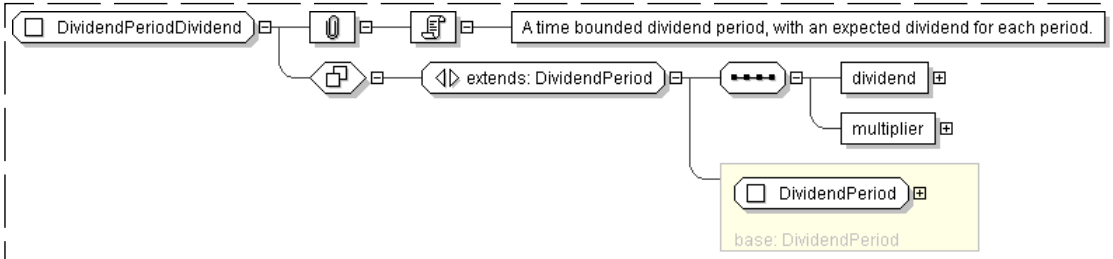
Super-types:	DividendPeriod < **DividendPeriodDividend** (by extension)
Sub-types:	None

Name	DividendPeriodDividend
Used by (from the same schema document)	Complex Type **DividendAdjustment**
Abstract	no
Documentation	A time bounded dividend period, with an expected dividend for each period.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <unadjustedStartDate> IdentifiedDate </unadjustedStartDate> [1]
   'Unadjusted inclusive dividend period start date.'

   <unadjustedEndDate> IdentifiedDate </unadjustedEndDate> [1]
   'Unadjusted inclusive dividend period end date.'

   <dateAdjustments> BusinessDayAdjustments </dateAdjustments> [1]
   'Date adjustments for all unadjusted dates in this dividend period.'

   <underlyerReference> AssetReference </underlyerReference> [0..1]
   'Reference to the underlyer which is paying dividends. This should be used in all cases,
   and must be used where there are multiple underlying assets, to avoid any ambiguity about
   which asset the dividend period relates to.'

   <dividend> Money </dividend> [1]
   'Expected dividend in this period.'

   <multiplier> PositiveDecimal </multiplier> [1]
   'Multiplier is a percentage value which is used to produce Deviation by multiplying
   the difference between Expected Dividend and Actual Dividend Deviation = Multiplier *
   (Expected Dividend – Actual Dividend).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="DividendPeriodDividend">
   <xsd:complexContent>
      <xsd:extension base=" DividendPeriod ">
         <xsd:sequence>
            <xsd:element name="dividend" type=" Money "/>
            <xsd:element name="multiplier" type=" PositiveDecimal "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
```

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

top

Complex Type: EquityCorporateEvents

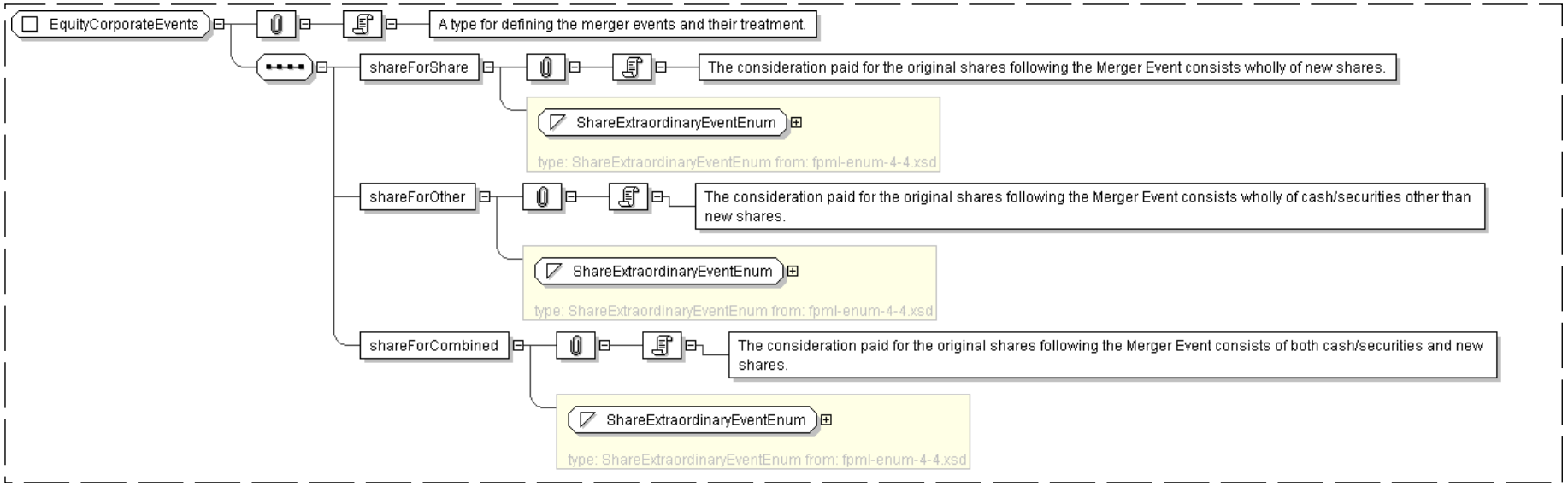
Super-types:	None
Sub-types:	None

Name	EquityCorporateEvents
Used by (from the same schema document)	Complex Type **ExtraordinaryEvents** , Complex Type **ExtraordinaryEvents**
Abstract	no
Documentation	A type for defining the merger events and their treatment.

XML Instance Representation

```
<...>
   <shareForShare> ShareExtraordinaryEventEnum </shareForShare> [1]
   'The consideration paid for the original shares following the Merger Event consists wholly
   of new shares.'

   <shareForOther> ShareExtraordinaryEventEnum </shareForOther> [1]
   'The consideration paid for the original shares following the Merger Event consists wholly
   of cash/securities other than new shares.'

   <shareForCombined> ShareExtraordinaryEventEnum </shareForCombined> [1]
   'The consideration paid for the original shares following the Merger Event consists of
   both cash/securities and new shares.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityCorporateEvents">
   <xsd:sequence>
      <xsd:element name="shareForShare" type=" ShareExtraordinaryEventEnum "/>
      <xsd:element name="shareForOther" type=" ShareExtraordinaryEventEnum "/>
```

```
      <xsd:element name="shareForCombined" type=" ShareExtraordinaryEventEnum "/>
    </xsd:sequence>
  </xsd:complexType>
```

top

Complex Type: EquityPremium

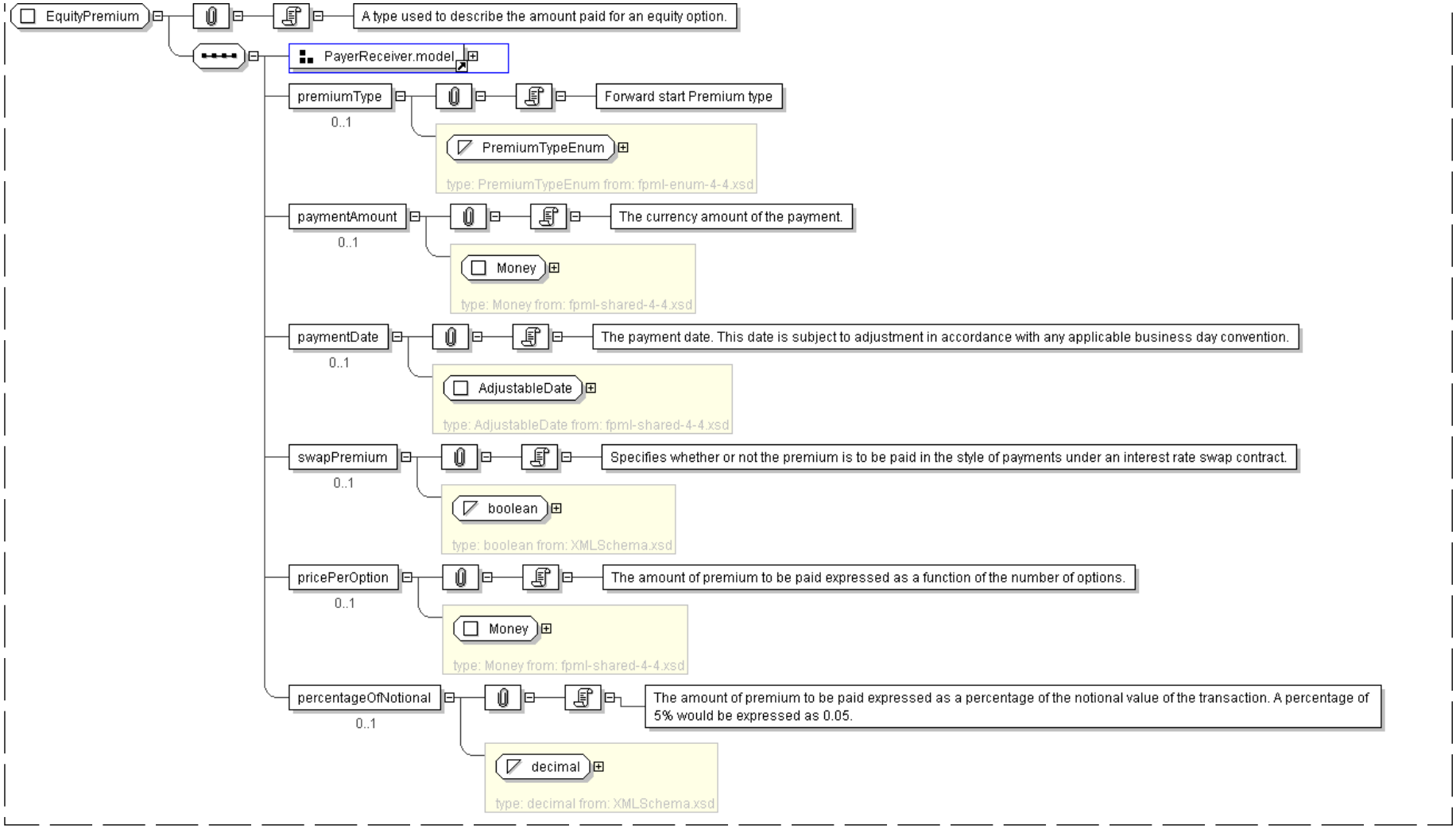
Super-types:	None
Sub-types:	None

Name	EquityPremium
Abstract	no
Documentation	A type used to describe the amount paid for an equity option.

XML Instance Representation

```
<...>
  <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
  'A reference to the party responsible for making the payments defined by this structure.'

  <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
  'A reference to the party that receives the payments corresponding to this structure.'

  <premiumType> PremiumTypeEnum </premiumType> [0..1]
  'Forward start Premium type'

  <paymentAmount> Money </paymentAmount> [0..1]
  'The currency amount of the payment.'

  <paymentDate> AdjustableDate </paymentDate> [0..1]
  'The payment date. This date is subject to adjustment in accordance with any
  applicable business day convention.'

  <swapPremium> xsd:boolean </swapPremium> [0..1]
  'Specifies whether or not the premium is to be paid in the style of payments under an
  interest rate swap contract.'

  <pricePerOption> Money </pricePerOption> [0..1]
  'The amount of premium to be paid expressed as a function of the number of options.'

  <percentageOfNotional> xsd:decimal </percentageOfNotional> [0..1]
  'The amount of premium to be paid expressed as a percentage of the notional value of
  the transaction. A percentage of 5% would be expressed as 0.05.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityPremium">
   <xsd:sequence>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="premiumType" type=" PremiumTypeEnum " minOccurs="0"/>
      <xsd:element name="paymentAmount" type=" Money " minOccurs="0"/>
      <xsd:element name="paymentDate" type=" AdjustableDate " minOccurs="0"/>
      <xsd:element name="swapPremium" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="pricePerOption" type=" Money " minOccurs="0"/>
      <xsd:element name="percentageOfNotional" type=" xsd:decimal " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: EquityStrike

Super-types:	None

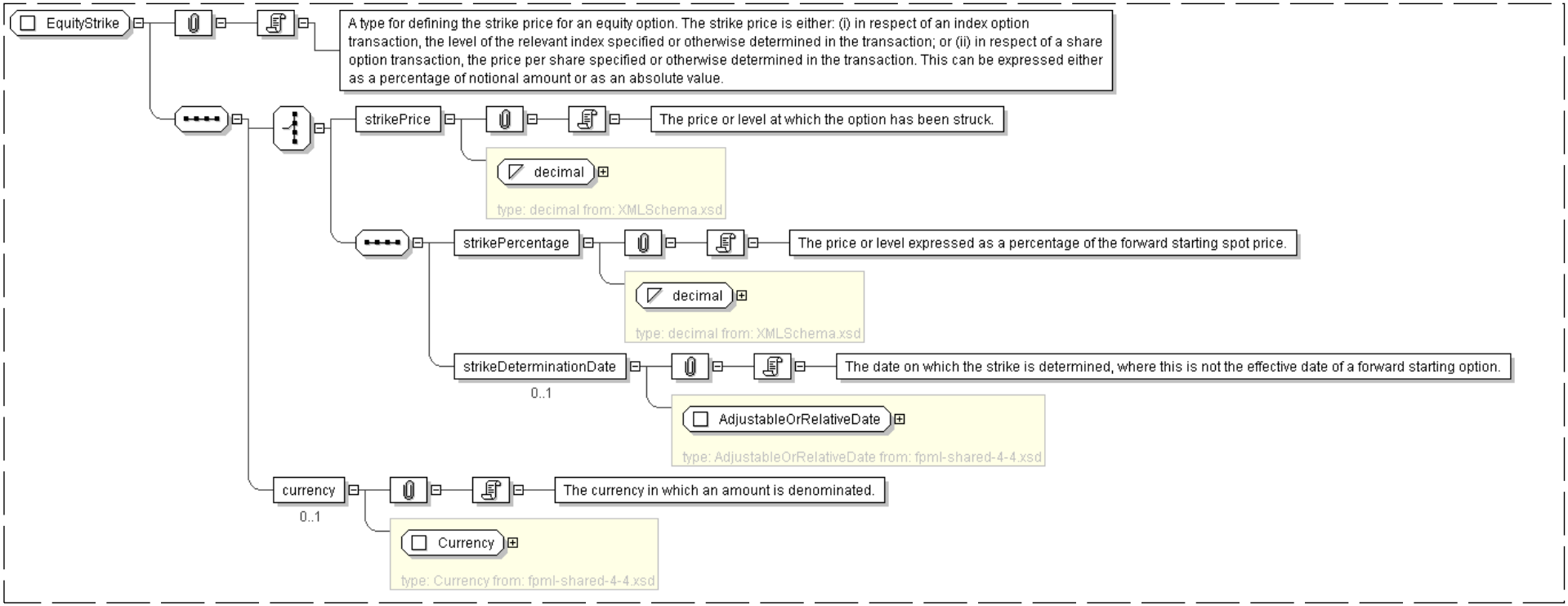
Sub-types:	None

Name	EquityStrike
Abstract	no
Documentation	A type for defining the strike price for an equity option. The strike price is either: (i) in respect of an index option transaction, the level of the relevant index specified or otherwise determined in the transaction; or (ii) in respect of a share option transaction, the price per share specified or otherwise determined in the transaction. This can be expressed either as a percentage of notional amount or as an absolute value.

XML Instance Representation

```
<...>
Start Choice [1]
      <strikePrice> xsd:decimal </strikePrice> [1]
      'The price or level at which the option has been struck.'

      <strikePercentage> xsd:decimal </strikePercentage> [1]
      'The price or level expressed as a percentage of the forward starting spot price.'

      <strikeDeterminationDate> AdjustableOrRelativeDate </strikeDeterminationDate> [0..1]
      'The date on which the strike is determined, where this is not the effective date of a
      forward starting option.'

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

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityStrike">
   <xsd:sequence>
```

```
      <xsd:choice>
         <xsd:element name="strikePrice" type=" xsd:decimal "/>
         <xsd:sequence>
            <xsd:element name="strikePercentage" type=" xsd:decimal "/>
            <xsd:element name="strikeDeterminationDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
         </xsd:sequence>
      </xsd:choice>
      <xsd:element name="currency" type=" Currency " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: EquityValuation

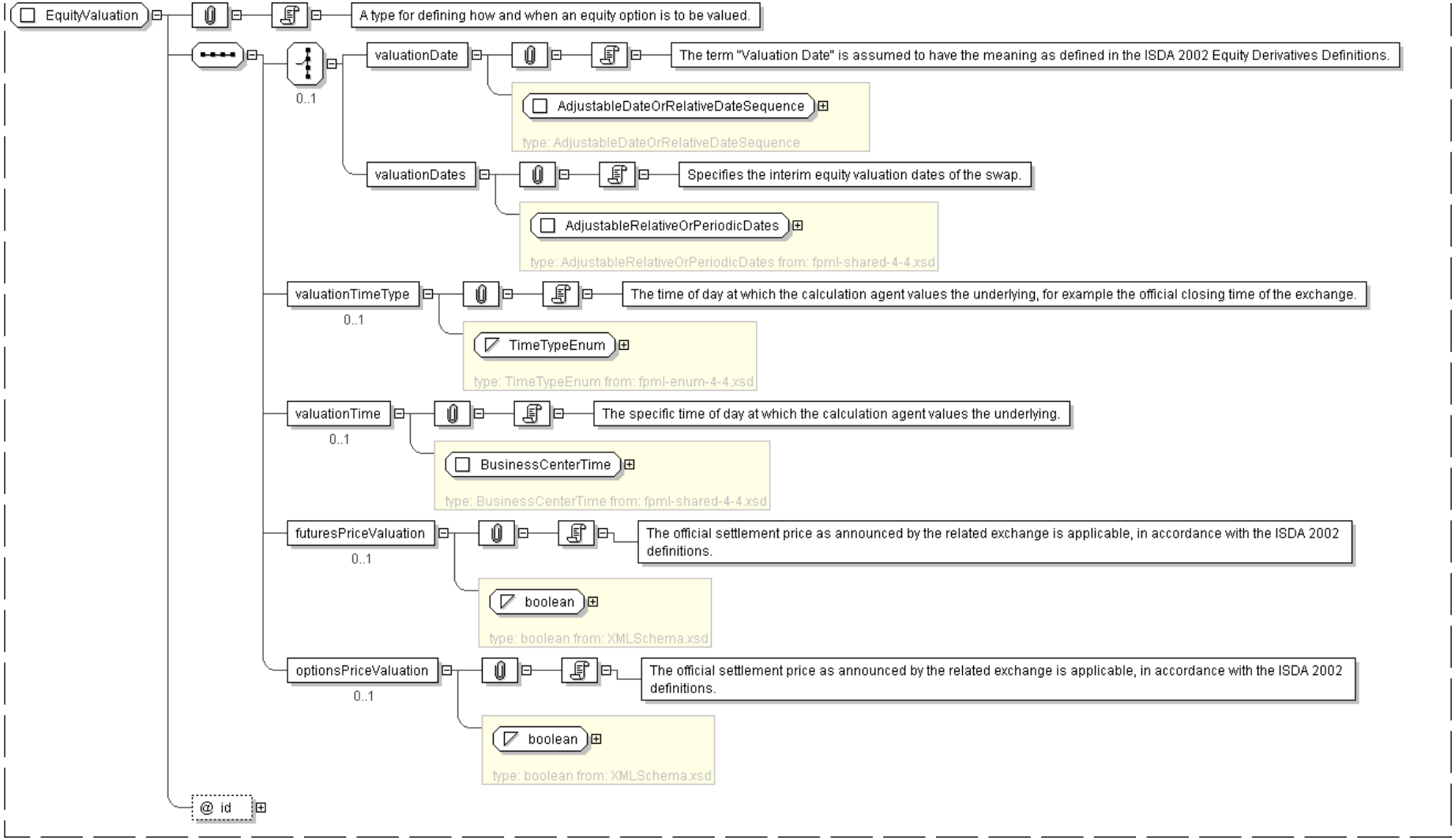
Super-types:	None
Sub-types:	None

Name	EquityValuation
Used by (from the same schema document)	Complex Type **DeprecatedVarianceLeg** , Complex Type **DirectionalLegUnderlyerValuation** , Complex Type **ReturnLegValuationPrice**
Abstract	no
Documentation	A type for defining how and when an equity option is to be valued.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
Start Choice [0..1]
      <valuationDate> AdjustableDateOrRelativeDateSequence </valuationDate> [1]
      'The term \"Valuation Date\" is assumed to have the meaning as defined in the ISDA 2002
      Equity Derivatives Definitions.'

      <valuationDates> AdjustableRelativeOrPeriodicDates </valuationDates> [1]
      'Specifies the interim equity valuation dates of the swap.'

End Choice
   <valuationTimeType> TimeTypeEnum </valuationTimeType> [0..1]
   'The time of day at which the calculation agent values the underlying, for example the
   official closing time of the exchange.'

   <valuationTime> BusinessCenterTime </valuationTime> [0..1]
   'The specific time of day at which the calculation agent values the underlying.'

   <futuresPriceValuation> xsd:boolean </futuresPriceValuation> [0..1]
   'The official settlement price as announced by the related exchange is applicable,
   in accordance with the ISDA 2002 definitions.'

   <optionsPriceValuation> xsd:boolean </optionsPriceValuation> [0..1]
   'The official settlement price as announced by the related exchange is applicable,
   in accordance with the ISDA 2002 definitions.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="EquityValuation">
   <xsd:sequence>
      <xsd:choice minOccurs="0">
         <xsd:element name="valuationDate" type=" AdjustableDateOrRelativeDateSequence "/>
         <xsd:element name="valuationDates" type=" AdjustableRelativeOrPeriodicDates "/>
      </xsd:choice>
      <xsd:element name="valuationTimeType" type=" TimeTypeEnum " minOccurs="0"/>
      <xsd:element name="valuationTime" type=" BusinessCenterTime " minOccurs="0"/>
      <xsd:element name="futuresPriceValuation" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="optionsPriceValuation" type=" xsd:boolean " minOccurs="0"/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

top

Complex Type: ExtraordinaryEvents

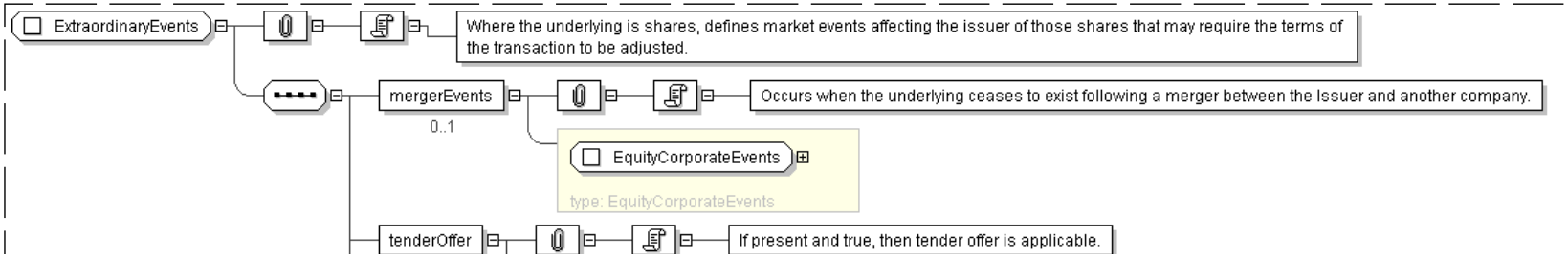
Super-types:	None
Sub-types:	None

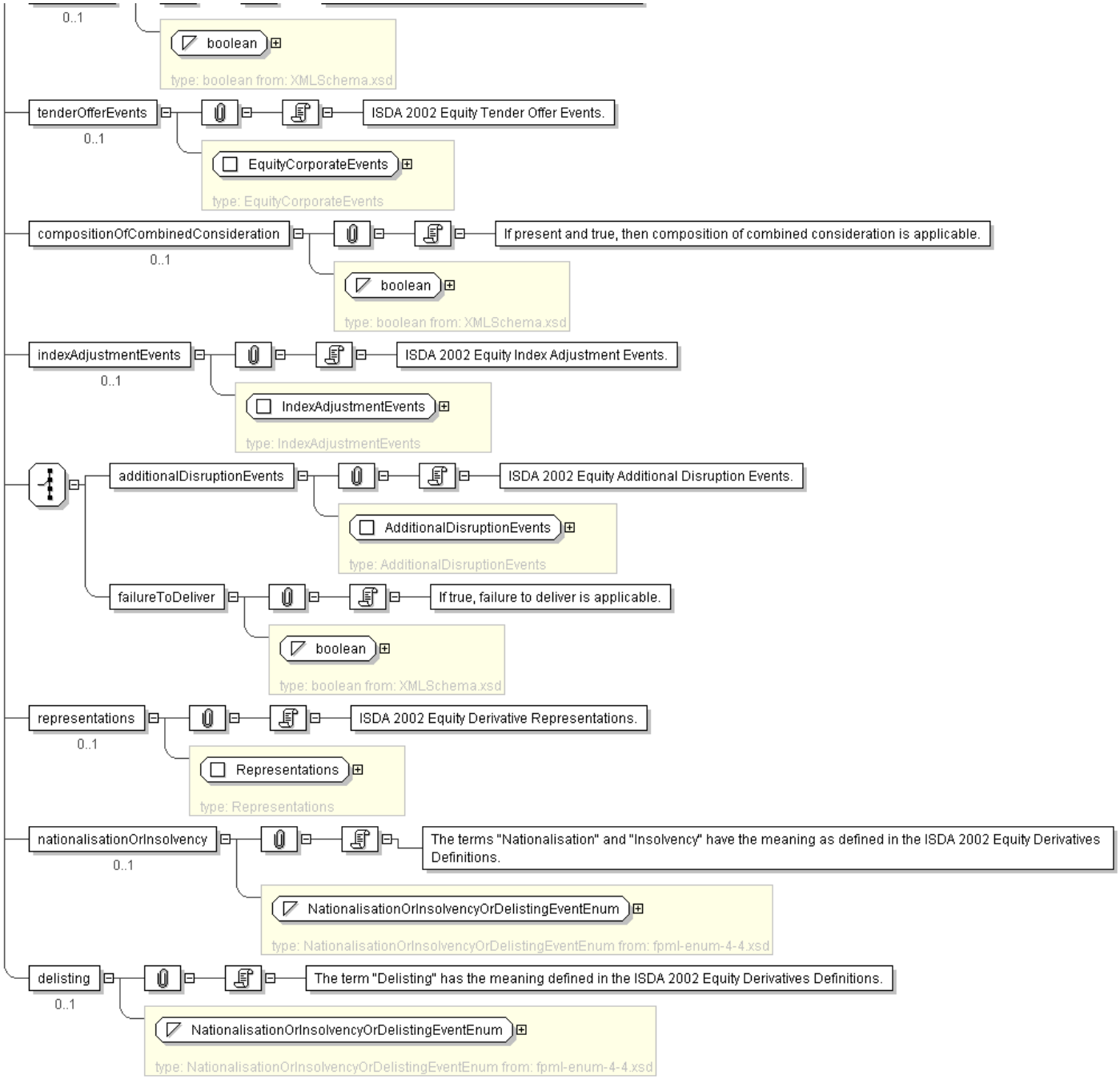
Name	ExtraordinaryEvents
Used by (from the same schema document)	Complex Type **NettedSwapBase** , Complex Type **ReturnSwap**
Abstract	no
Documentation	Where the underlying is shares, defines market events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.

XML Instance Representation

```
<...>
   <mergerEvents> EquityCorporateEvents </mergerEvents> [0..1]
   'Occurs when the underlying ceases to exist following a merger between the Issuer and
   another company.'

   <tenderOffer> xsd:boolean </tenderOffer> [0..1]
   'If present and true, then tender offer is applicable.'

   <tenderOfferEvents> EquityCorporateEvents </tenderOfferEvents> [0..1]
   'ISDA 2002 Equity Tender Offer Events.'

   <compositionOfCombinedConsideration> xsd:boolean </compositionOfCombinedConsideration> [0..1]
   'If present and true, then composition of combined consideration is applicable.'

   <indexAdjustmentEvents> IndexAdjustmentEvents </indexAdjustmentEvents> [0..1]
   'ISDA 2002 Equity Index Adjustment Events.'

Start Choice [1]
      <additionalDisruptionEvents> AdditionalDisruptionEvents </additionalDisruptionEvents> [1]
      'ISDA 2002 Equity Additional Disruption Events.'

      <failureToDeliver> xsd:boolean </failureToDeliver> [1]
      'If true, failure to deliver is applicable.'

End Choice
   <representations> Representations </representations> [0..1]
   'ISDA 2002 Equity Derivative Representations.'

   <nationalisationOrInsolvency> NationalisationOrInsolvencyOrDelistingEventEnum
   </nationalisationOrInsolvency> [0..1]
   'The terms \"Nationalisation\" and \"Insolvency\" have the meaning as defined in the ISDA
   2002 Equity Derivatives Definitions.'

   <delisting> NationalisationOrInsolvencyOrDelistingEventEnum </delisting> [0..1]
   'The term \"Delisting\" has the meaning defined in the ISDA 2002 Equity
   Derivatives Definitions.'

</...>
```

Diagram





Schema Component Representation

```
<xsd:complexType name="ExtraordinaryEvents">
   <xsd:sequence>
      <xsd:element name="mergerEvents" type=" EquityCorporateEvents " minOccurs="0"/>
      <xsd:element name="tenderOffer" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="tenderOfferEvents" type=" EquityCorporateEvents " minOccurs="0"/>
```

```
      <xsd:element name="compositionOfCombinedConsideration" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="indexAdjustmentEvents" type=" IndexAdjustmentEvents " minOccurs="0"/>
      <xsd:choice>
         <xsd:element name="additionalDisruptionEvents" type=" AdditionalDisruptionEvents "/>
         <xsd:element name="failureToDeliver" type=" xsd:boolean "/>
      </xsd:choice>
      <xsd:element name="representations" type=" Representations " minOccurs="0"/>
      <xsd:element name="nationalisationOrInsolvency"
      type=" NationalisationOrInsolvencyOrDelistingEventEnum " minOccurs="0"/>
      <xsd:element name="delisting" type=" NationalisationOrInsolvencyOrDelistingEventEnum
      " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: IndexAdjustmentEvents

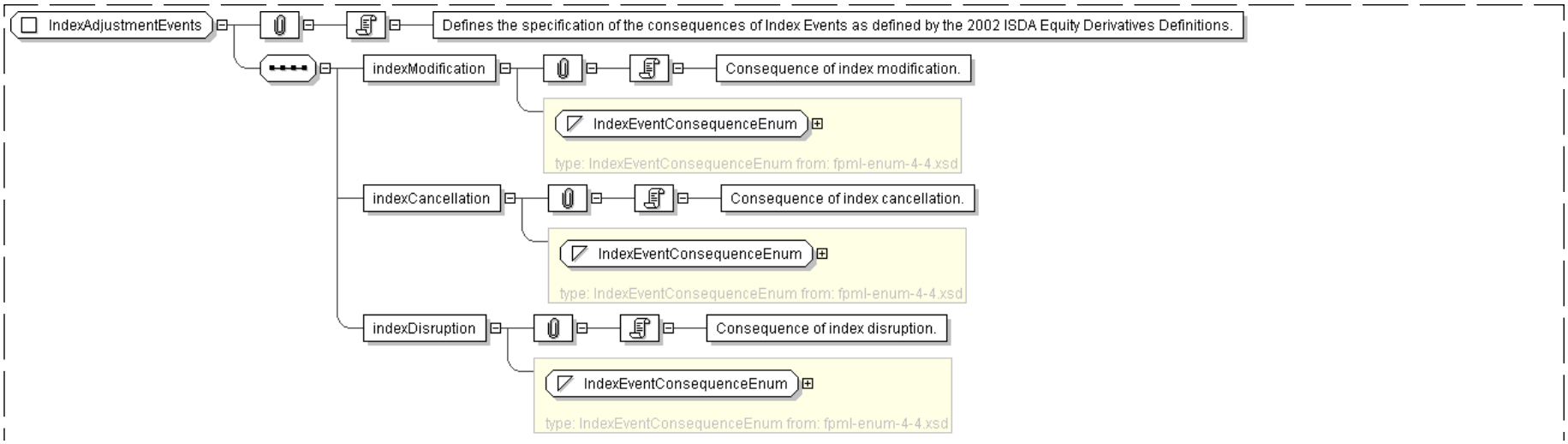
Super-types:	None
Sub-types:	None

Name	IndexAdjustmentEvents
Used by (from the same schema document)	Complex Type **ExtraordinaryEvents**
Abstract	no
Documentation	Defines the specification of the consequences of Index Events as defined by the 2002 ISDA Equity Derivatives Definitions.

XML Instance Representation

```
<...>
   <indexModification> IndexEventConsequenceEnum </indexModification> [1]
   'Consequence of index modification.'

   <indexCancellation> IndexEventConsequenceEnum </indexCancellation> [1]
   'Consequence of index cancellation.'

   <indexDisruption> IndexEventConsequenceEnum </indexDisruption> [1]
   'Consequence of index disruption.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="IndexAdjustmentEvents">
   <xsd:sequence>
      <xsd:element name="indexModification" type=" IndexEventConsequenceEnum "/>
      <xsd:element name="indexCancellation" type=" IndexEventConsequenceEnum "/>
      <xsd:element name="indexDisruption" type=" IndexEventConsequenceEnum "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: InterestCalculation

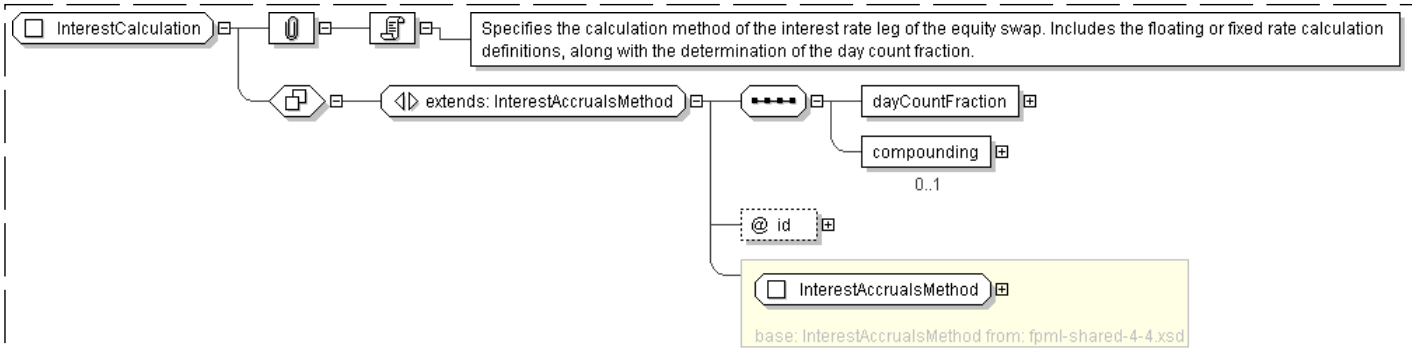
Super-types:	InterestAccrualsMethod < **InterestCalculation** (by extension)
Sub-types:	None

Name	InterestCalculation
Used by (from the same schema document)	Complex Type **InterestLeg**
Abstract	no
Documentation	Specifies the calculation method of the interest rate leg of the equity swap. Includes the floating or fixed rate calculation definitions, along with the determination of the day count fraction.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
Start Choice [1]
      <floatingRateCalculation> FloatingRateCalculation </floatingRateCalculation> [1]
      'The floating rate calculation definitions'

      <fixedRate> xsd:decimal </fixedRate> [1]
      'The calculation period fixed rate. A per annum rate, expressed as a decimal. A fixed rate
      of 5% would be represented as 0.05.'

End Choice
   <dayCountFraction> DayCountFraction </dayCountFraction> [1]
   'The day count fraction.'

   <compounding> Compounding </compounding> [0..1]
   'Defines compounding rates on the Interest Leg.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestCalculation">
   <xsd:complexContent>
      <xsd:extension base=" InterestAccrualsMethod ">
         <xsd:sequence>
            <xsd:element name="dayCountFraction" type=" DayCountFraction "/>
            <xsd:element name="compounding" type=" Compounding " minOccurs="0"/>
         </xsd:sequence>
         <xsd:attribute name="id" type=" xsd:ID "/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: InterestCalculationReference

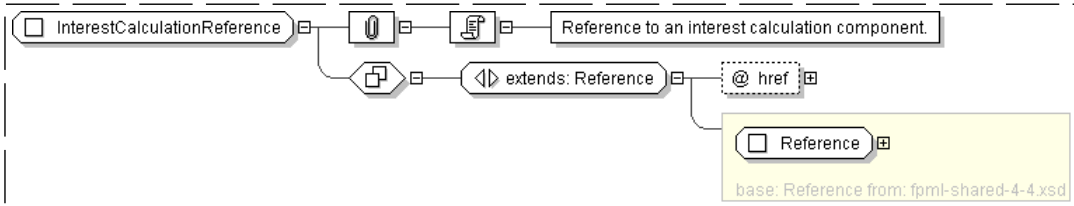
Super-types:	Reference < **InterestCalculationReference** (by extension)
Sub-types:	None

Name	InterestCalculationReference
Used by (from the same schema document)	Complex Type **CompoundingRate**
Abstract	no
Documentation	Reference to an interest calculation component.

XML Instance Representation

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

Diagram



Schema Component Representation

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

top

Complex Type: InterestLeg

Super-types:	Leg < ReturnSwapLeg (by extension) < **InterestLeg** (by extension)
Sub-types:	None

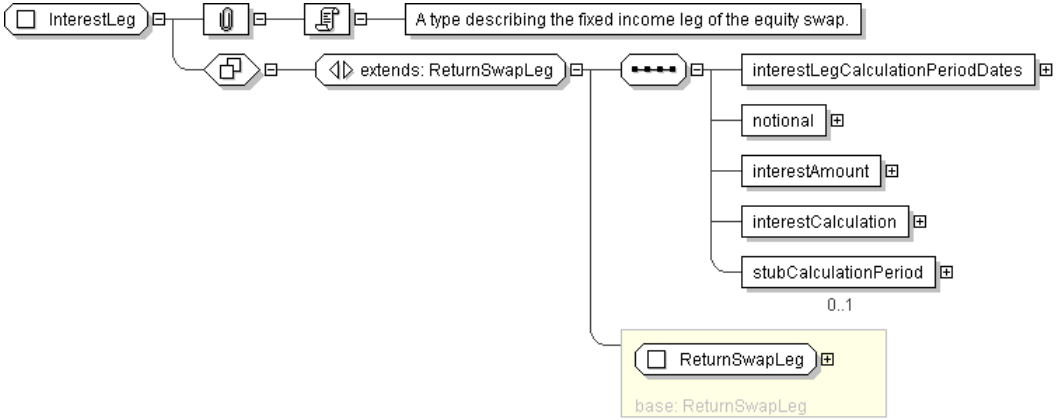
Name	InterestLeg
Used by (from the same schema document)	Element **interestLeg**
Abstract	no

Documentation A type describing the fixed income leg of the equity swap.

XML Instance Representation

```
<...
legIdentifier=" xsd:ID [0..1]
  'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
     'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
     'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
     'DEPRECATED This element will be removed in the next FpML major version. Frequency at
     which this leg pays.'

   <interestLegCalculationPeriodDates> InterestLegCalculationPeriodDates
   </interestLegCalculationPeriodDates> [1]
     'Component that holds the various dates used to specify the interest leg of the equity swap.
     It is used to define the InterestPeriodDates identifyer.'

   <notional> ReturnSwapNotional </notional> [1]
     'Specifies the notional of a return type swap. When used in the equity leg, the definition
     will typically combine the actual amount (using the notional component defined by the
     FpML industry group) and the determination method. When used in the interest leg,
     the definition will typically point to the definition of the equity leg.'

   <interestAmount> LegAmount </interestAmount> [1]
     'Specifies, in relation to each Interest Payment Date, the amount to which the Interest
     Payment Date relates. Unless otherwise specified, this term has the meaning defined in the
     ISDA 2000 ISDA Definitions.'

   <interestCalculation> InterestCalculation </interestCalculation> [1]
     'Specifies the calculation method of the interest rate leg of the equity swap. Includes
     the floating or fixed rate calculation definitions, along with the determination of the
     day count fraction.'

   <stubCalculationPeriod> StubCalculationPeriod </stubCalculationPeriod> [0..1]
     'Specifies the stub calculation period'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLeg">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="interestLegCalculationPeriodDates" type="
            InterestLegCalculationPeriodDates "/>
            <xsd:element name="notional" type=" ReturnSwapNotional "/>
            <xsd:element name="interestAmount" type=" LegAmount "/>
            <xsd:element name="interestCalculation" type=" InterestCalculation "/>
            <xsd:element name="stubCalculationPeriod" type=" StubCalculationPeriod " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: InterestLegCalculationPeriodDates

Super-types:	None
Sub-types:	None

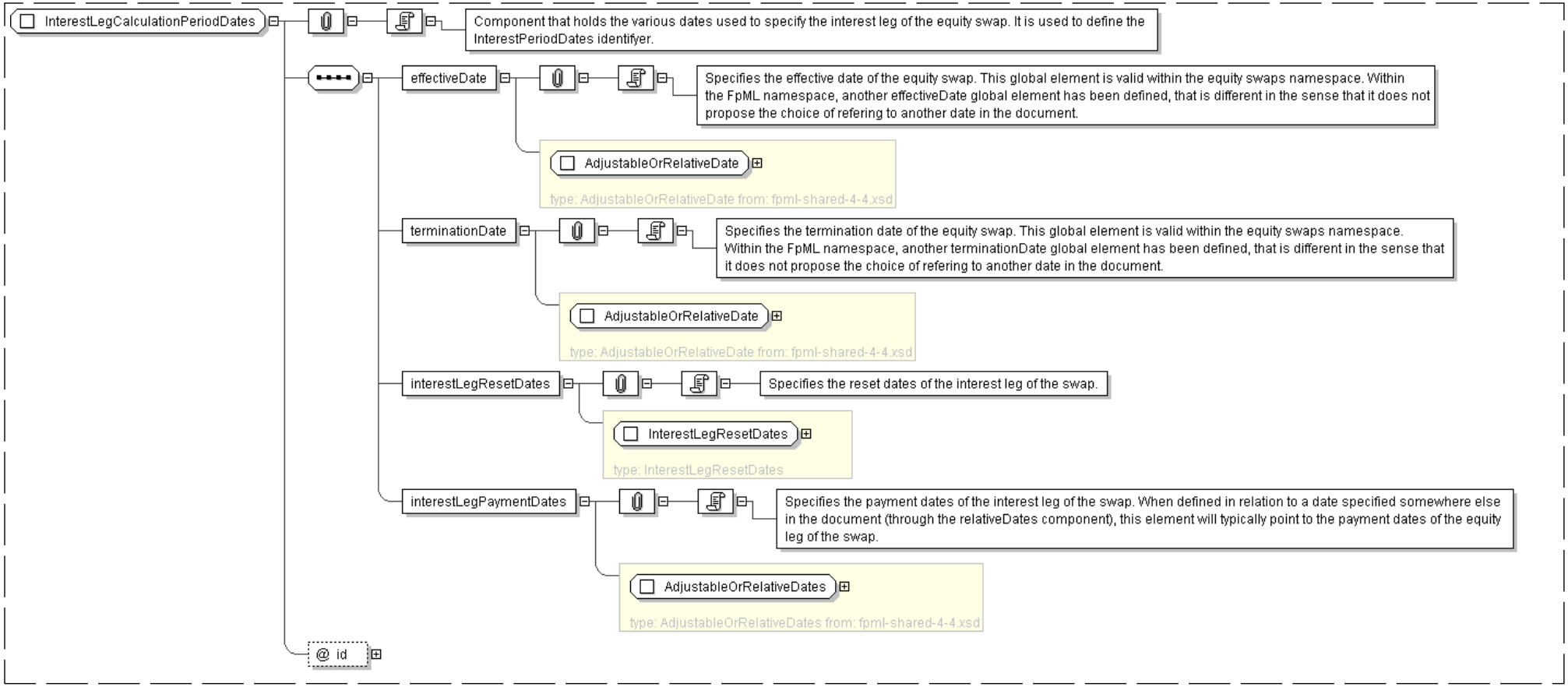
Name	InterestLegCalculationPeriodDates
Used by (from the same schema document)	Complex Type **InterestLeg**
Abstract	no
Documentation	Component that holds the various dates used to specify the interest leg of the equity swap. It is used to define the InterestPeriodDates identifyer.

XML Instance Representation

```
<...
 id=" xsd:ID [1]">
    <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
    'Specifies the effective date of the equity swap. This global element is valid within
    the equity swaps namespace. Within the FpML namespace, another effectiveDate global element
    has been defined, that is different in the sense that it does not propose the choice
    of refering to another date in the document.'

    <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
    'Specifies the termination date of the equity swap. This global element is valid within
    the equity swaps namespace. Within the FpML namespace, another terminationDate global
    element has been defined, that is different in the sense that it does not propose the choice
    of refering to another date in the document.'
```

```
<interestLegResetDates> InterestLegResetDates </interestLegResetDates> [1]
'Specifies the reset dates of the interest leg of the swap.'

<interestLegPaymentDates> AdjustableOrRelativeDates </interestLegPaymentDates> [1]
'Specifies the payment dates of the interest leg of the swap. When defined in relation to
a date specified somewhere else in the document (through the relativeDates component),
this element will typically point to the payment dates of the equity leg of the swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLegCalculationPeriodDates">
   <xsd:sequence>
      <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
      <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
      <xsd:element name="interestLegResetDates" type=" InterestLegResetDates "/>
      <xsd:element name="interestLegPaymentDates" type=" AdjustableOrRelativeDates "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID " use="required"/>
</xsd:complexType>
```

top

Complex Type: InterestLegCalculationPeriodDatesReference

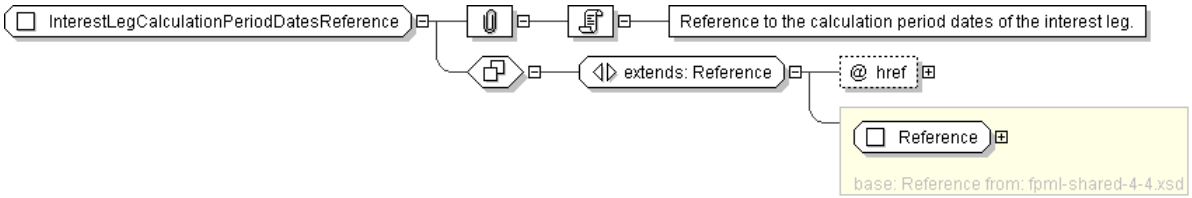
Super-types:	Reference < **InterestLegCalculationPeriodDatesReference** (by extension)
Sub-types:	None

Name	InterestLegCalculationPeriodDatesReference
Used by (from the same schema document)	Complex Type **InterestLegResetDates**
Abstract	no
Documentation	Reference to the calculation period dates of the interest leg.

XML Instance Representation

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

Diagram



Schema Component Representation

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

top

Complex Type: InterestLegResetDates

Super-types:	None
Sub-types:	None

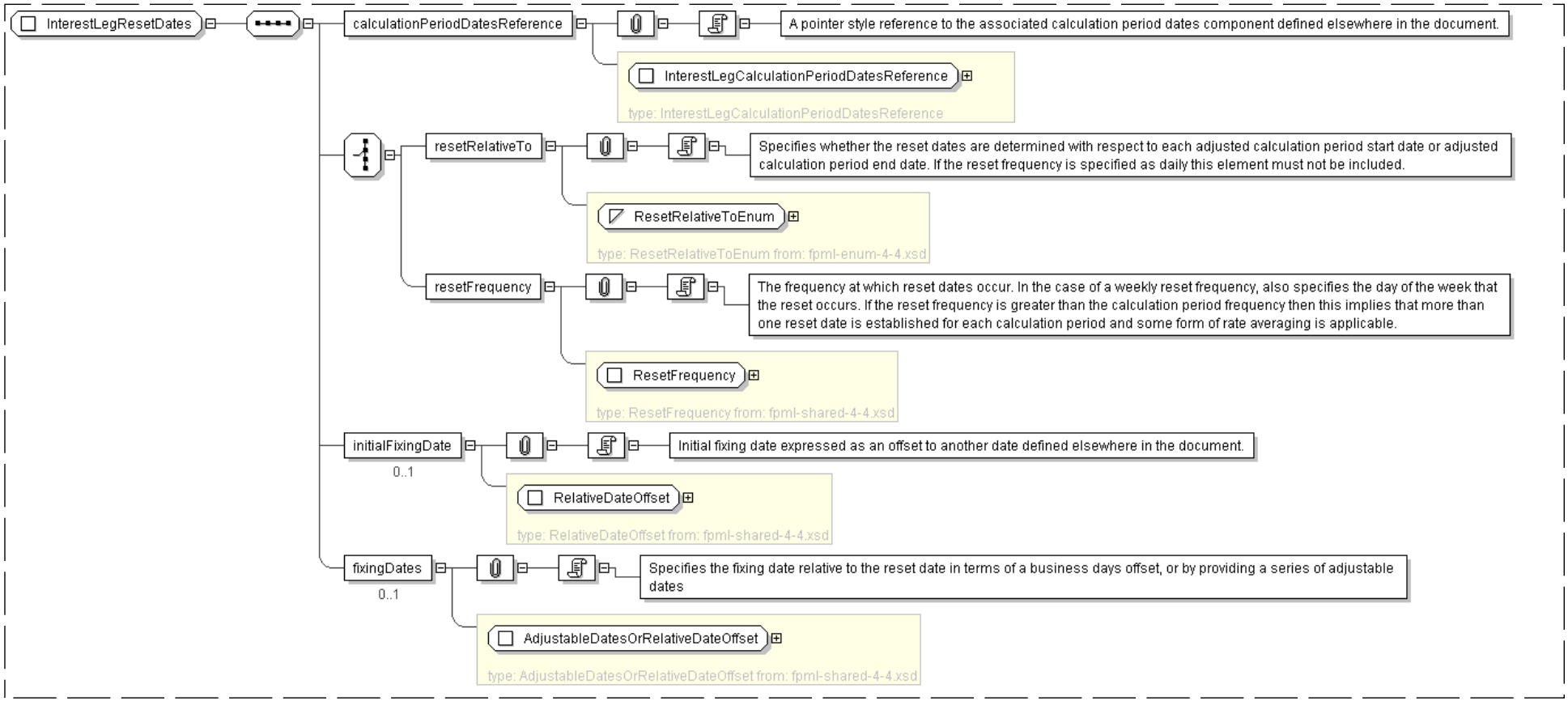
Name	InterestLegResetDates
Used by (from the same schema document)	Complex Type **InterestLegCalculationPeriodDates**
Abstract	no

XML Instance Representation

```
<...>
   <calculationPeriodDatesReference> InterestLegCalculationPeriodDatesReference
   </calculationPeriodDatesReference> [1]
   'A pointer style reference to the associated calculation period dates component
   defined elsewhere in the document.'

Start Choice [1]
      <resetRelativeTo> ResetRelativeToEnum </resetRelativeTo> [1]
      'Specifies whether the reset dates are determined with respect to each adjusted
      calculation period start date or adjusted calculation period end date. If the reset
      frequency is specified as daily this element must not be included.'

      <resetFrequency> ResetFrequency </resetFrequency> [1]
      'The frequency at which reset dates occur. In the case of a weekly reset frequency,
```

```
      also specifies the day of the week that the reset occurs. If the reset frequency is
      greater than the calculation period frequency then this implies that more than one reset
      date is established for each calculation period and some form of rate averaging is applicable.'

End Choice
   <initialFixingDate> RelativeDateOffset </initialFixingDate> [0..1]
   'Initial fixing date expressed as an offset to another date defined elsewhere in the document.'

   <fixingDates> AdjustableDatesOrRelativeDateOffset </fixingDates> [0..1]
   'Specifies the fixing date relative to the reset date in terms of a business days offset, or
   by providing a series of adjustable dates'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="InterestLegResetDates">
   <xsd:sequence>
      <xsd:element name="calculationPeriodDatesReference"
      type=" InterestLegCalculationPeriodDatesReference "/>
      <xsd:choice>
         <xsd:element name="resetRelativeTo" type=" ResetRelativeToEnum "/>
         <xsd:element name="resetFrequency" type=" ResetFrequency "/>
      </xsd:choice>
      <xsd:element name="initialFixingDate" type=" RelativeDateOffset " minOccurs="0"/>
      <xsd:element name="fixingDates" type=" AdjustableDatesOrRelativeDateOffset " minOccurs="0"/>
   </xsd:sequence>
```

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

top

Complex Type: LegAmount

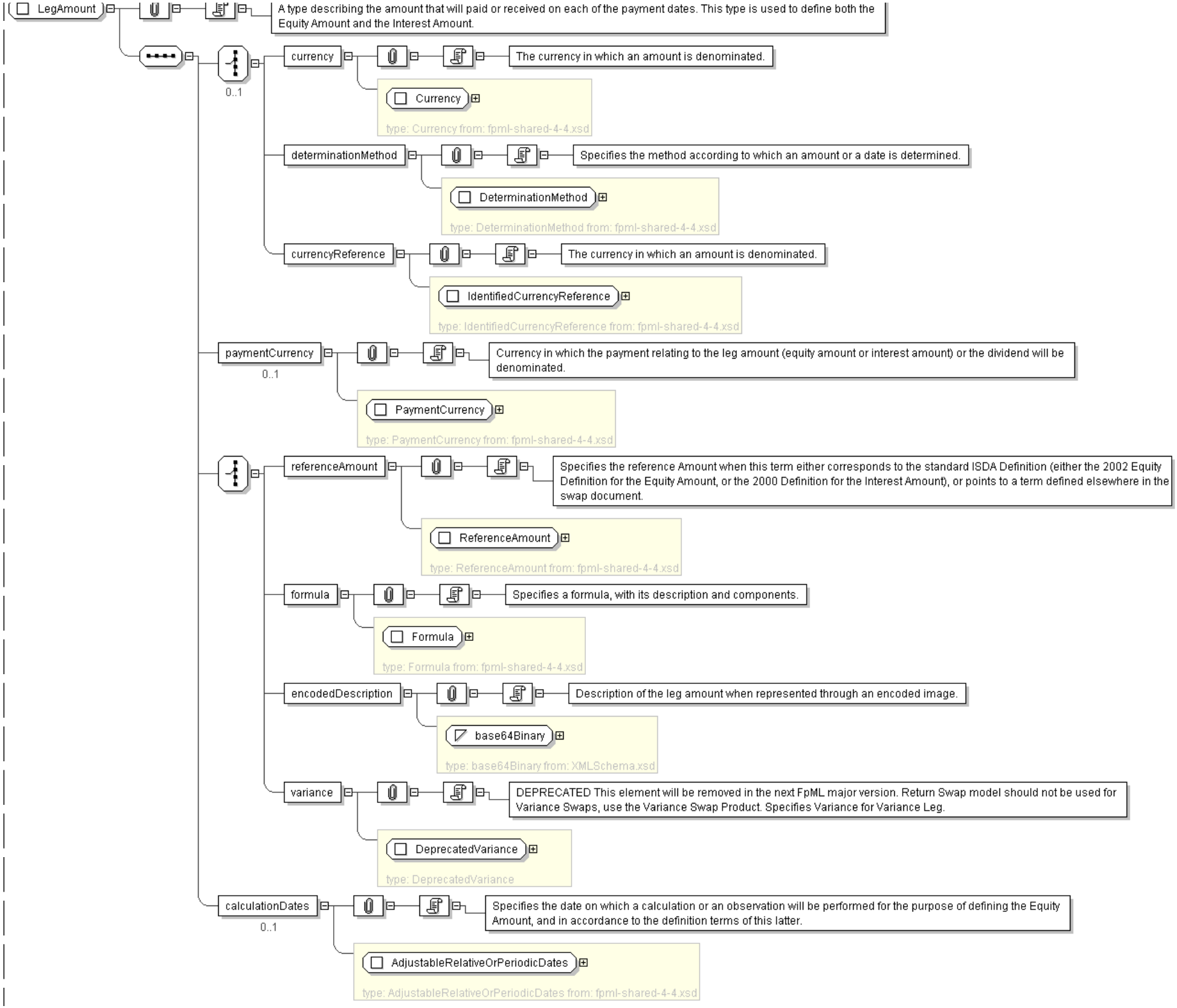
Super-types:	None
Sub-types:	• ReturnSwapAmount (by extension) ○ DeprecatedVarianceAmount (by extension)

Name	LegAmount
Used by (from the same schema document)	Complex Type **InterestLeg**
Abstract	no
Documentation	A type describing the amount that will paid or received on each of the payment dates. This type is used to define both the Equity Amount and the Interest Amount.

XML Instance Representation

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

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

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

End Choice
   <paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]
   'Currency in which the payment relating to the leg amount (equity amount or interest amount)
   or the dividend will be denominated.'

Start Choice [1]
      <referenceAmount> ReferenceAmount </referenceAmount> [1]
      'Specifies the reference Amount when this term either corresponds to the standard
      ISDA Definition (either the 2002 Equity Definition for the Equity Amount, or the
      2000 Definition for the Interest Amount), or points to a term defined elsewhere in the
      swap document.'

      <formula> Formula </formula> [1]
      'Specifies a formula, with its description and components.'

      <encodedDescription> xsd:base64Binary </encodedDescription> [1]
      'Description of the leg amount when represented through an encoded image.'

      <variance> DeprecatedVariance </variance> [1]
      'DEPRECATED This element will be removed in the next FpML major version. Return Swap
      model should not be used for Variance Swaps, use the Variance Swap Product. Specifies
      Variance for Variance Leg.'

End Choice
   <calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]
   'Specifies the date on which a calculation or an observation will be performed for the
   purpose of defining the Equity Amount, and in accordance to the definition terms of
   this latter.'

</...>
```

Diagram


LegAmount
A type describing the amount that will paid or received on each of the payment dates. This type is used to define both the Equity Amount and the Interest Amount.
0..1
currency
The currency in which an amount is denominated.
Currency
type: Currency from: fpml-shared-4-4.xsd
determinationMethod
Specifies the method according to which an amount or a date is determined.
DeterminationMethod
type: DeterminationMethod from: fpml-shared-4-4.xsd
currencyReference
The currency in which an amount is denominated.
IdentifiedCurrencyReference
type: IdentifiedCurrencyReference from: fpml-shared-4-4.xsd
paymentCurrency
0..1
Currency in which the payment relating to the leg amount (equity amount or interest amount) or the dividend will be denominated.
PaymentCurrency
type: PaymentCurrency from: fpml-shared-4-4.xsd
referenceAmount
Specifies the reference Amount when this term either corresponds to the standard ISDA Definition (either the 2002 Equity Definition for the Equity Amount, or the 2000 Definition for the Interest Amount), or points to a term defined elsewhere in the swap document.
ReferenceAmount
type: ReferenceAmount from: fpml-shared-4-4.xsd
formula
Specifies a formula, with its description and components.
Formula
type: Formula from: fpml-shared-4-4.xsd
encodedDescription
Description of the leg amount when represented through an encoded image.
base64Binary
type: base64Binary from: XMLSchema.xsd
variance
DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. Specifies Variance for Variance Leg.
DeprecatedVariance
type: DeprecatedVariance
calculationDates
0..1
Specifies the date on which a calculation or an observation will be performed for the purpose of defining the Equity Amount, and in accordance to the definition terms of this latter.
AdjustableRelativeOrPeriodicDates
type: AdjustableRelativeOrPeriodicDates from: fpml-shared-4-4.xsd

Schema Component Representation

```
<xsd:complexType name="LegAmount">
   <xsd:sequence>
      <xsd:choice minOccurs="0">
         <xsd:element name="currency" type=" Currency "/>
         <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
         <xsd:element name="currencyReference" type=" IdentifiedCurrencyReference "/>
      </xsd:choice>
      <xsd:element name="paymentCurrency" type=" PaymentCurrency " minOccurs="0"
      deprecated="true" deprecatedReason="The model is wrong since it has an intradocument
      reference that is not clear. Current PaymentCurrency model and elements using this type
      are deprecated. Instead, the choice above between currency, determinationMethod,
      and currencyReference (of type CurrencyReference) should be used."/>
      <xsd:choice>
         <xsd:element name="referenceAmount" type=" ReferenceAmount "/>
         <xsd:element name="formula" type=" Formula "/>
         <xsd:element name="encodedDescription" type=" xsd:base64Binary "/>
         <xsd:element name="variance" type=" DeprecatedVariance "
         deprecated="true" deprecatedReason="Return Swap model should not be used for Variance
         Swaps, use the Variance Swap Product"/>
      </xsd:choice>
      <xsd:element name="calculationDates" type=" AdjustableRelativeOrPeriodicDates " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: MakeWholeProvisions

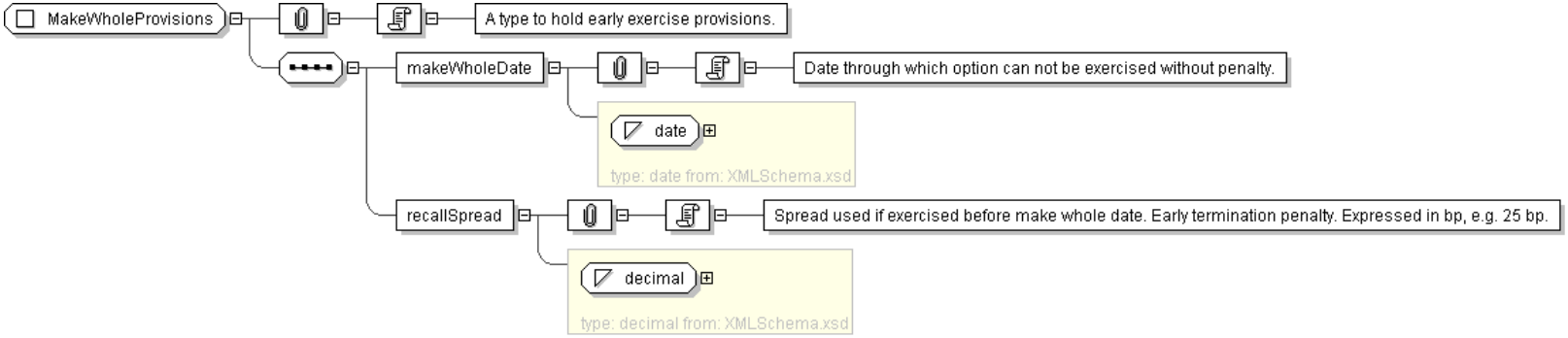
Super-types:	None
Sub-types:	None

Name	MakeWholeProvisions
Abstract	no
Documentation	A type to hold early exercise provisions.

XML Instance Representation

```
<...>
   <makeWholeDate> xsd:date </makeWholeDate> [1]
   'Date through which option can not be exercised without penalty.'

   <recallSpread> xsd:decimal </recallSpread> [1]
   'Spread used if exercised before make whole date. Early termination penalty. Expressed in bp,
   e.g. 25 bp.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="MakeWholeProvisions">
   <xsd:sequence>
      <xsd:element name="makeWholeDate" type=" xsd:date "/>
      <xsd:element name="recallSpread" type=" xsd:decimal "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: NettedSwapBase

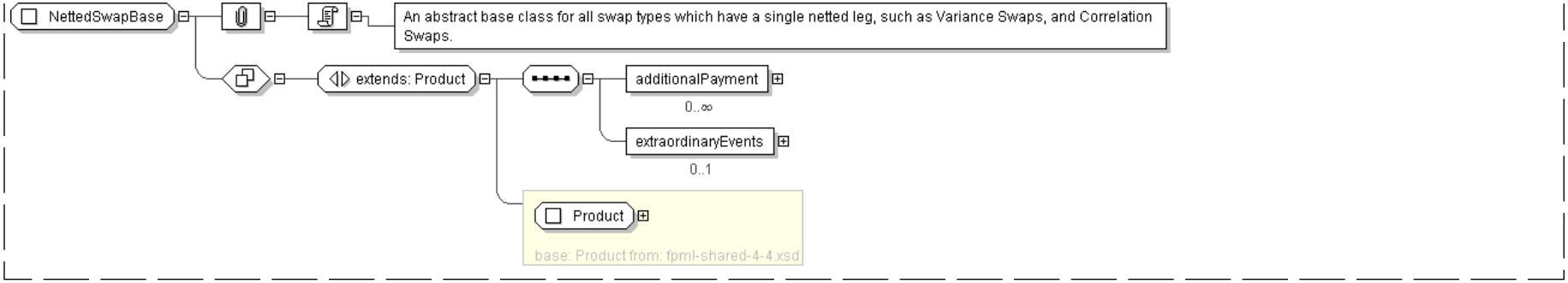
Super-types:	Product < **NettedSwapBase** (by extension)
Sub-types:	None

Name	NettedSwapBase
Abstract	yes
Documentation	An abstract base class for all swap types which have a single netted leg, such as Variance Swaps, and Correlation Swaps.

XML Instance Representation

```
<...
 id=" xsd:ID [0..1]">
    <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using
    a coding scheme.'

    <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain
    values associated with this element. Note that the domain values for this element are
    not strictly an enumerated list.'

    <additionalPayment> ClassifiedPayment </additionalPayment> [0..*]
    'Specifies additional payment(s) between the principal parties to the netted swap.'

    <extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]
    'Where the underlying is shares, specifies events affecting the issuer of those shares that
    may require the terms of the transaction to be adjusted.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="NettedSwapBase" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:element name="additionalPayment" type=" ClassifiedPayment "
            minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: OptionFeatures

Super-types:	None
Sub-types:	None

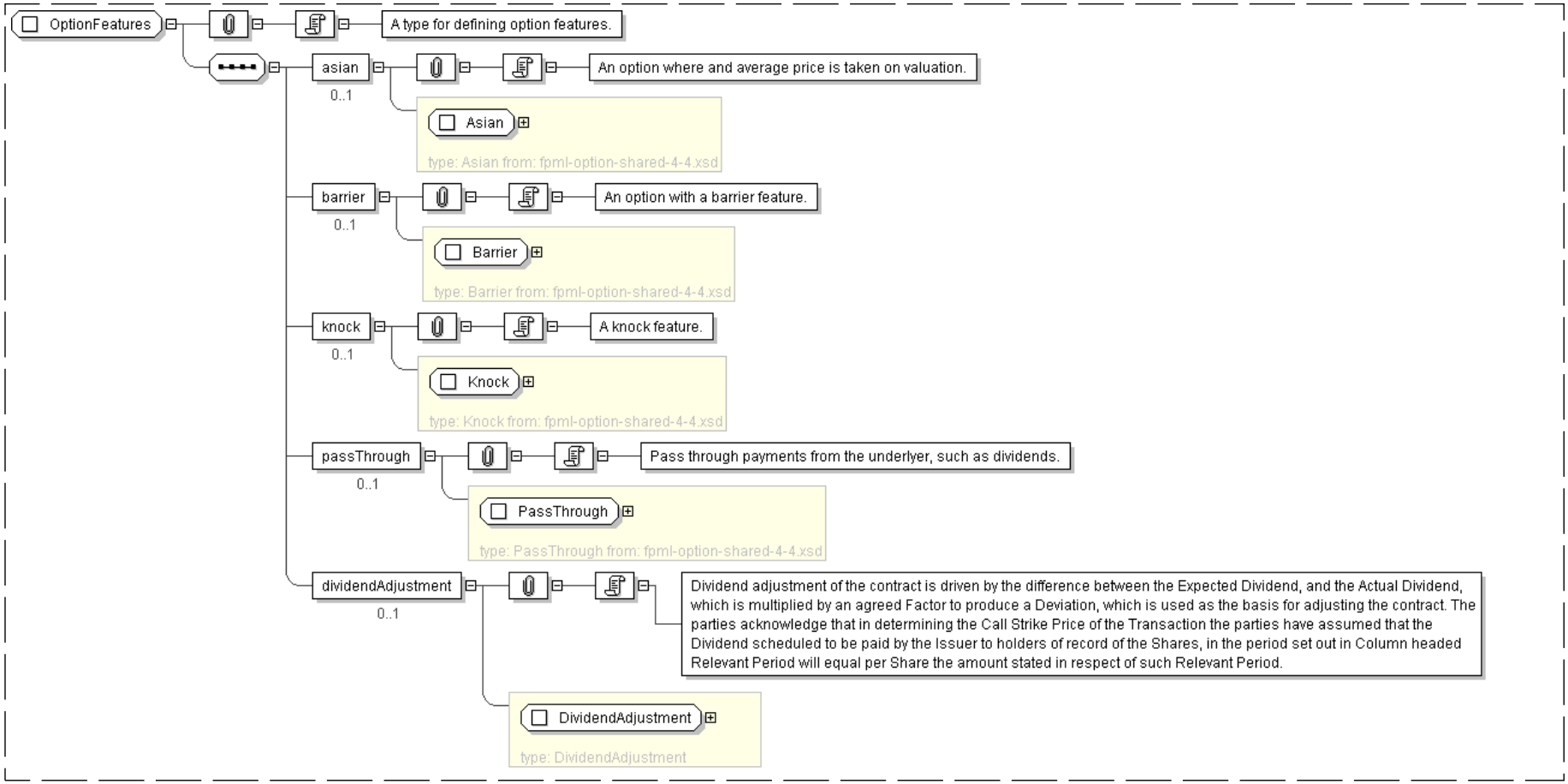
Name	OptionFeatures
Used by (from the same schema document)	Model Group **Feature.model**
Abstract	no
Documentation	A type for defining option features.

XML Instance Representation

```
<...>
   <asian> Asian </asian> [0..1]
   'An option where and average price is taken on valuation.'

   <barrier> Barrier </barrier> [0..1]
   'An option with a barrier feature.'

   <knock> Knock </knock> [0..1]
   'A knock feature.'

   <passThrough> PassThrough </passThrough> [0..1]
   'Pass through payments from the underlyer, such as dividends.'

   <dividendAdjustment> DividendAdjustment </dividendAdjustment> [0..1]
   'Dividend adjustment of the contract is driven by the difference between the Expected
   Dividend, and the Actual Dividend, which is multiplied by an agreed Factor to produce
   a Deviation, which is used as the basis for adjusting the contract. The parties
   acknowledge that in determining the Call Strike Price of the Transaction the parties
   have assumed that the Dividend scheduled to be paid by the Issuer to holders of record of
   the Shares, in the period set out in Column headed Relevant Period will equal per Share
   the amount stated in respect of such Relevant Period.'
```

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

Diagram



Schema Component Representation

```
<xsd:complexType name="OptionFeatures">
   <xsd:sequence>
      <xsd:element name="asian" type=" Asian " minOccurs="0"/>
      <xsd:element name="barrier" type=" Barrier " minOccurs="0"/>
      <xsd:element name="knock" type=" Knock " minOccurs="0"/>
      <xsd:element name="passThrough" type=" PassThrough " minOccurs="0"/>
      <xsd:element name="dividendAdjustment" type=" DividendAdjustment " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: PrincipalExchangeAmount

Super-types:	None
Sub-types:	None

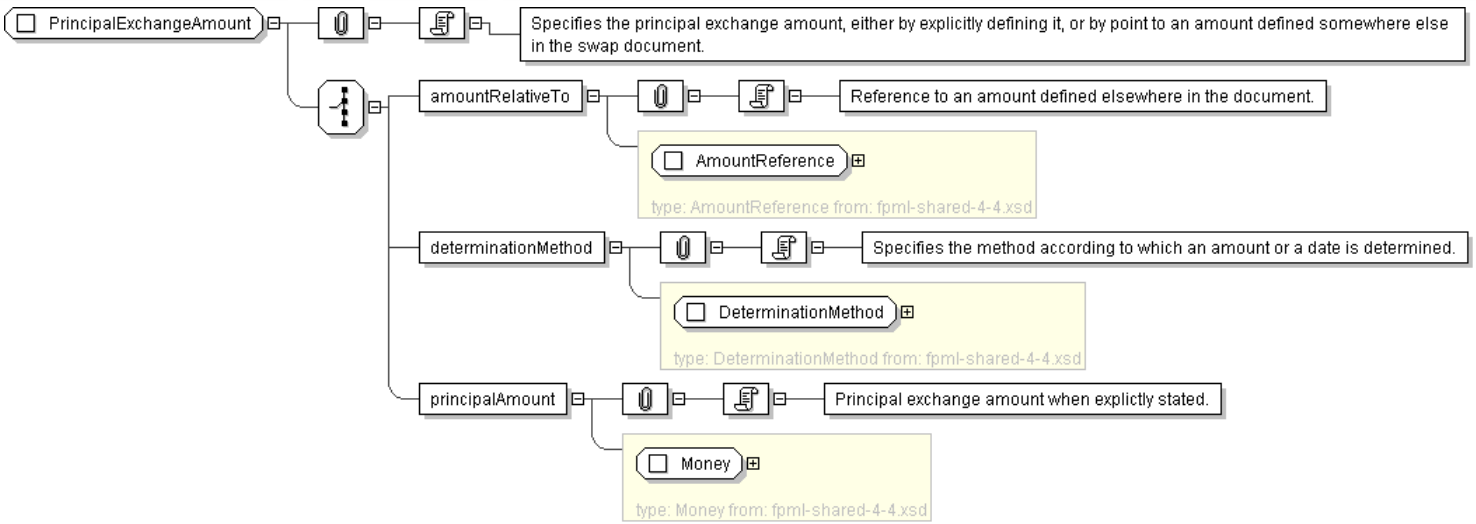
Name	PrincipalExchangeAmount

Used by (from the same schema document)	Complex Type **PrincipalExchangeDescriptions**
Abstract	no
Documentation	Specifies the principal exchange amount, either by explicitly defining it, or by point to an amount defined somewhere else in the swap document.

XML Instance Representation

```
<...>
Start Choice [1]
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'Reference to an amount defined elsewhere in the document.'

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <principalAmount> Money </principalAmount> [1]
      'Principal exchange amount when explictly stated.'

End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeAmount">
   <xsd:choice>
      <xsd:element name="amountRelativeTo" type=" AmountReference "/>
      <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
      <xsd:element name="principalAmount" type=" Money "/>
   </xsd:choice>
</xsd:complexType>
```

top

Complex Type: PrincipalExchangeDescriptions

Super-types:	None
Sub-types:	None

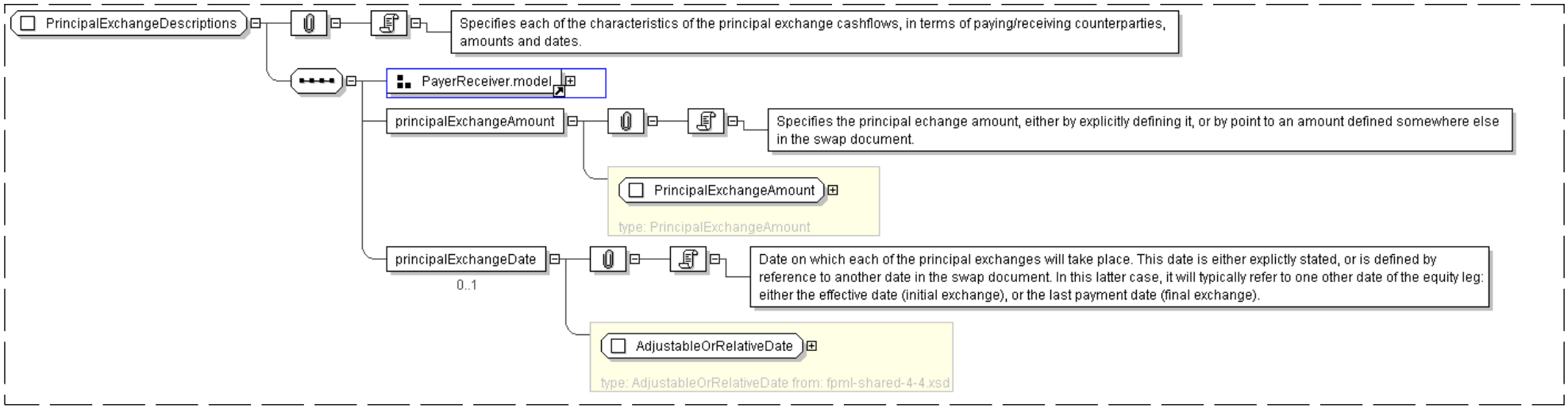
Name	PrincipalExchangeDescriptions

Used by (from the same schema document)	Complex Type **PrincipalExchangeFeatures**
Abstract	no
Documentation	Specifies each of the characteristics of the principal exchange cashflows, in terms of paying/receiving counterparties, amounts and dates.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <principalExchangeAmount> PrincipalExchangeAmount </principalExchangeAmount> [1]
   'Specifies the principal echange amount, either by explicitly defining it, or by point to
   an amount defined somewhere else in the swap document.'

   <principalExchangeDate> AdjustableOrRelativeDate </principalExchangeDate> [0..1]
   'Date on which each of the principal exchanges will take place. This date is either
   explictly stated, or is defined by reference to another date in the swap document. In
   this latter case, it will typically refer to one other date of the equity leg: either
   the effective date (initial exchange), or the last payment date (final exchange).'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeDescriptions">
   <xsd:sequence>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="principalExchangeAmount" type=" PrincipalExchangeAmount "/>
      <xsd:element name="principalExchangeDate" type=" AdjustableOrRelativeDate " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: PrincipalExchangeFeatures

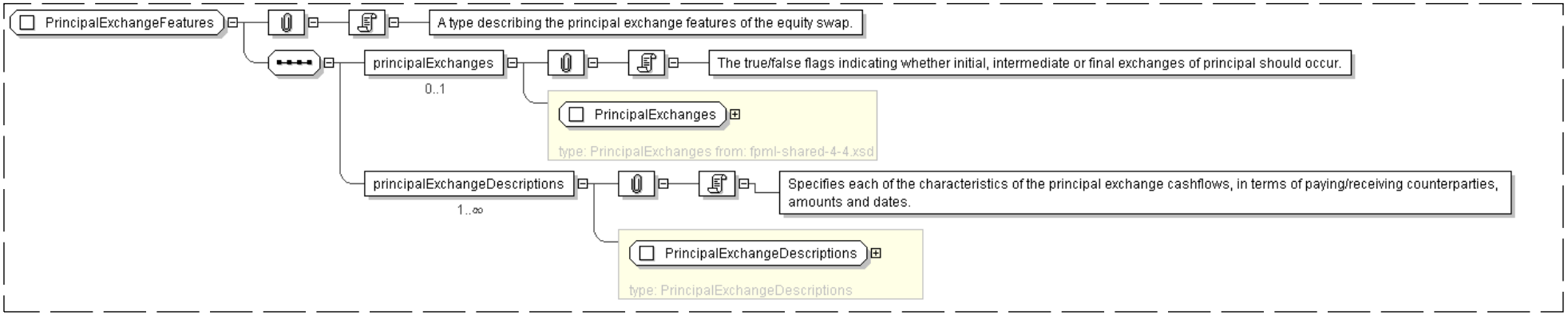
Super-types:	None
Sub-types:	None

Name	PrincipalExchangeFeatures
Used by (from the same schema document)	Complex Type **ReturnSwapBase**
Abstract	no
Documentation	A type describing the principal exchange features of the equity swap.

XML Instance Representation

```
<...>
   <principalExchanges> PrincipalExchanges </principalExchanges> [0..1]
   'The true/false flags indicating whether initial, intermediate or final exchanges of
   principal should occur.'

   <principalExchangeDescriptions> PrincipalExchangeDescriptions </
   principalExchangeDescriptions> [1..*]
   'Specifies each of the characteristics of the principal exchange cashflows, in terms of
   paying/receiving counterparties, amounts and dates.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="PrincipalExchangeFeatures">
   <xsd:sequence>
      <xsd:element name="principalExchanges" type=" PrincipalExchanges " minOccurs="0"/>
      <xsd:element name="principalExchangeDescriptions" type=" PrincipalExchangeDescriptions
      " maxOccurs="unbounded"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: Representations

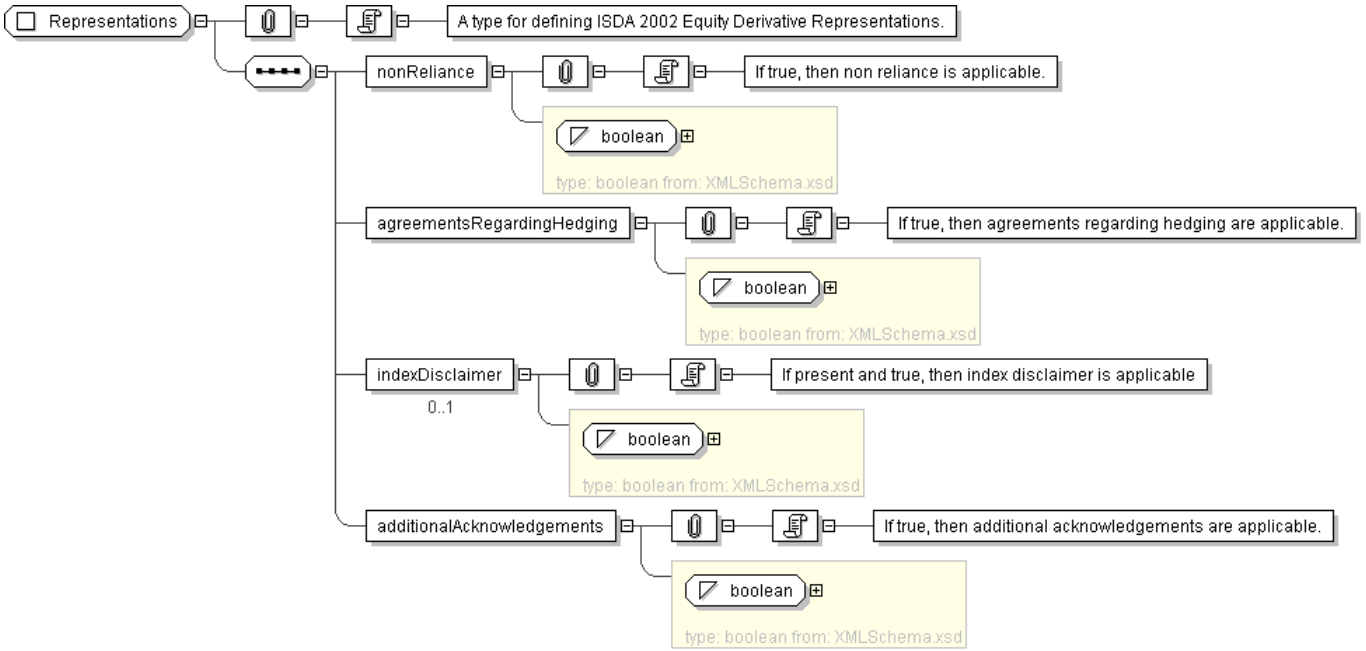
Super-types:	None
Sub-types:	None

Name	Representations
Used by (from the same schema document)	Complex Type **ExtraordinaryEvents**
Abstract	no
Documentation	A type for defining ISDA 2002 Equity Derivative Representations.

XML Instance Representation

```
<...>
   <nonReliance> xsd:boolean </nonReliance> [1]
   'If true, then non reliance is applicable.'

   <agreementsRegardingHedging> xsd:boolean </agreementsRegardingHedging> [1]
   'If true, then agreements regarding hedging are applicable.'

   <indexDisclaimer> xsd:boolean </indexDisclaimer> [0..1]
   'If present and true, then index disclaimer is applicable'

   <additionalAcknowledgements> xsd:boolean </additionalAcknowledgements> [1]
   'If true, then additional acknowledgements are applicable.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Representations">
   <xsd:sequence>
      <xsd:element name="nonReliance" type=" xsd:boolean "/>
      <xsd:element name="agreementsRegardingHedging" type=" xsd:boolean "/>
      <xsd:element name="indexDisclaimer" type=" xsd:boolean " minOccurs="0"/>
      <xsd:element name="additionalAcknowledgements" type=" xsd:boolean "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: Return

Super-types:	None

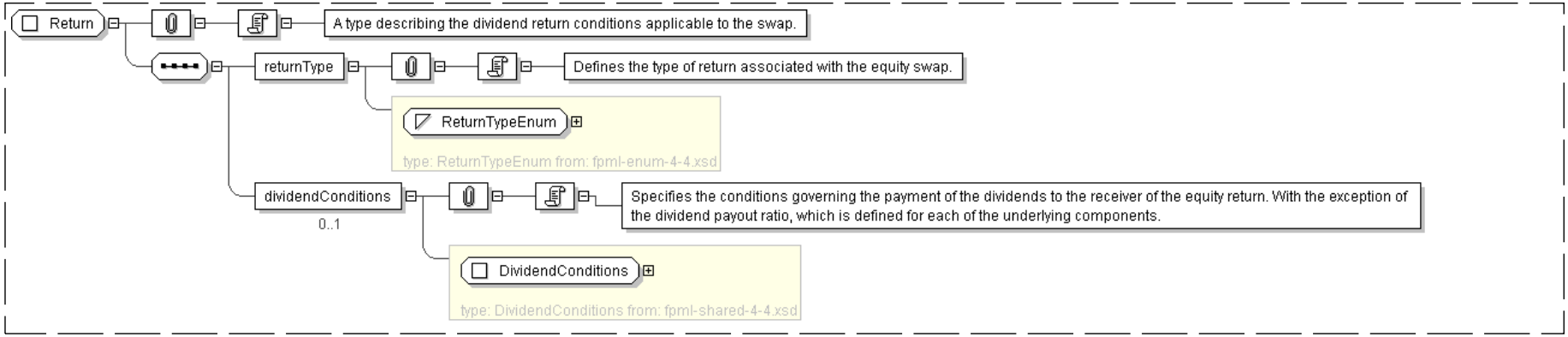
Sub-types:	None

Name	Return
Used by (from the same schema document)	Complex Type **ReturnLeg**
Abstract	no
Documentation	A type describing the dividend return conditions applicable to the swap.

XML Instance Representation

```
<...>
   <returnType> ReturnTypeEnum </returnType> [1]
   'Defines the type of return associated with the equity swap.'

   <dividendConditions> DividendConditions </dividendConditions> [0..1]
   'Specifies the conditions governing the payment of the dividends to the receiver of the
   equity return. With the exception of the dividend payout ratio, which is defined for each
   of the underlying components.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="Return">
   <xsd:sequence>
      <xsd:element name="returnType" type=" ReturnTypeEnum "/>
      <xsd:element name="dividendConditions" type=" DividendConditions " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: ReturnLeg

Super-types:	Leg < ReturnSwapLeg (by extension) < ReturnSwapLegUnderlyer (by extension) < **ReturnLeg** (by extension)
Sub-types:	None

Name	ReturnLeg
Used by (from the same schema document)	Element **returnLeg**
Abstract	no
Documentation	A type describing the return leg of a return type swap.

XML Instance Representation

```
<...
```

legIdentifier=" xsd:ID [0..1]

'DEPRECATED This element will be renamed to id in the next major FpML version.'

">

<payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]

'A reference to the party responsible for making the payments defined by this structure.'

<receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]

'A reference to the party that receives the payments corresponding to this structure.'

<paymentFrequency> Interval </paymentFrequency> [0..1]

'DEPRECATED This element will be removed in the next FpML major version. Frequency at which this leg pays.'

<effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]

'Specifies the effective date of this leg of the swap. When defined in relation to a date specified somewhere else in the document (through the relativeDate component), this element will typically point to the effective date of the other leg of the swap.'

<terminationDate> AdjustableOrRelativeDate </terminationDate> [1]

'Specifies the termination date of this leg of the swap. When defined in relation to a date specified somewhere else in the document (through the relativeDate component), this element will typically point to the termination date of the other leg of the swap.'

<underlyer> Underlyer </underlyer> [1]

'Specifies the underlying component of the leg, which can be either one or many and consists in either equity, index or convertible bond component, or a combination of these.'

<rateOfReturn> ReturnLegValuation </rateOfReturn> [1]

'Element named \"valuation\" in versions prior to FpML 4.2 Second Working Draft. Specifies the terms of the initial price of the return type swap and of the subsequent valuations of the underlyer.'

<notional> ReturnSwapNotional </notional> [1]

'Specifies the notional of a return type swap. When used in the equity leg, the definition will typically combine the actual amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest leg, the definition will typically point to the definition of the equity leg.'

<amount> ReturnSwapAmount </amount> [1]

'Element named \"equityAmount\" in versions prior to FpML 4.2 Second Working Draft. Specifies, in relation to each Payment Date, the amount to which the Payment Date relates. For equity swaps this element is equivalent to the Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.'

<return> Return </return> [1]

'Specifies the conditions under which dividend affecting the underlyer will be paid to the receiver of the amounts.'

<notionalAdjustments> NotionalAdjustmentEnum </notionalAdjustments> [1]

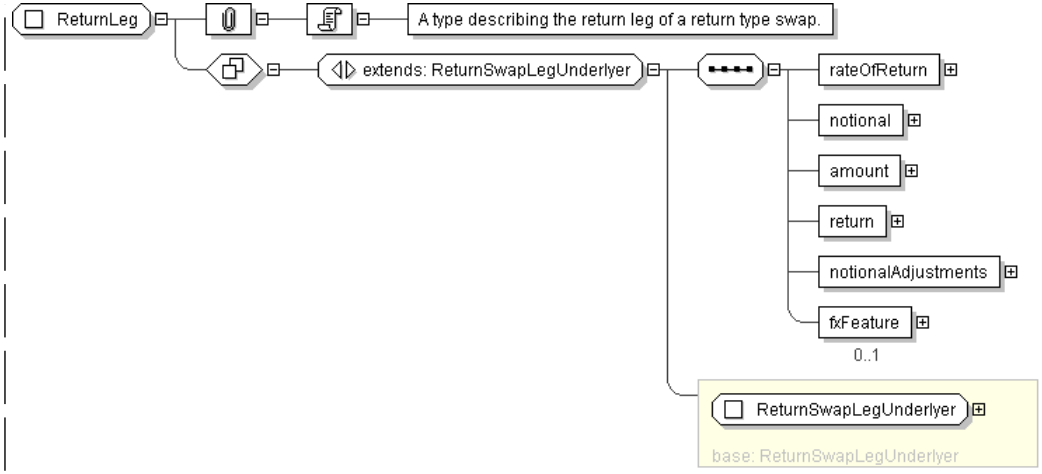
'Specifies the conditions that govern the adjustment to the number of units of the equity swap.'

<fxFeature> FxFeature </fxFeature> [0..1]

'A quanto or composite FX feature.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnLeg">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLegUnderlyer ">
         <xsd:sequence>
            <xsd:element name="rateOfReturn" type=" ReturnLegValuation "/>
            <xsd:element name="notional" type=" ReturnSwapNotional "/>
            <xsd:element name="amount" type=" ReturnSwapAmount "/>
            <xsd:element name="return" type=" Return "/>
            <xsd:element name="notionalAdjustments" type=" NotionalAdjustmentEnum "/>
            <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnLegValuation

Super-types:	None
Sub-types:	None

Name	ReturnLegValuation
Used by (from the same schema document)	Complex Type **ReturnLeg**
Abstract	no
Documentation	A type describing the initial and final valuation of the underlyer.

XML Instance Representation

```
<...>
   <initialPrice> ReturnLegValuationPrice </initialPrice> [1]
   'Specifies the initial reference price of the underlyer. This price can be expressed either
   as an actual amount/currency, as a determination method, or by reference to another
   value specified in the swap document.'

   <notionalReset> xsd:boolean </notionalReset> [1]
   'Element named \"equityNotionalReset\" in versions prior to FpML 4.2 Second Working Draft.
   For equity swaps, this element is equivalent to the term \"Equity Notional Reset\" as
   defined in the ISDA 2002 Equity Derivatives Definitions. The reference to the ISDA
   definition is either \"Applicable\" or \'Inapplicable\".'
```

```
<valuationPriceInterim> ReturnLegValuationPrice </valuationPriceInterim> [0..1]
```

'Specifies the interim valuation price of the underlyer. This price can be expressed either as an actual amount/currency, as a determination method, or by reference to another value specified in the swap document.'

```
<valuationPriceFinal> ReturnLegValuationPrice </valuationPriceFinal> [1]
```

'Specifies the final valuation price of the underlyer. This price can be expressed either as an actual amount/currency, as a determination method, or by reference to another value specified in the swap document.'

```
<paymentDates> ReturnSwapPaymentDates </paymentDates> [1]
```

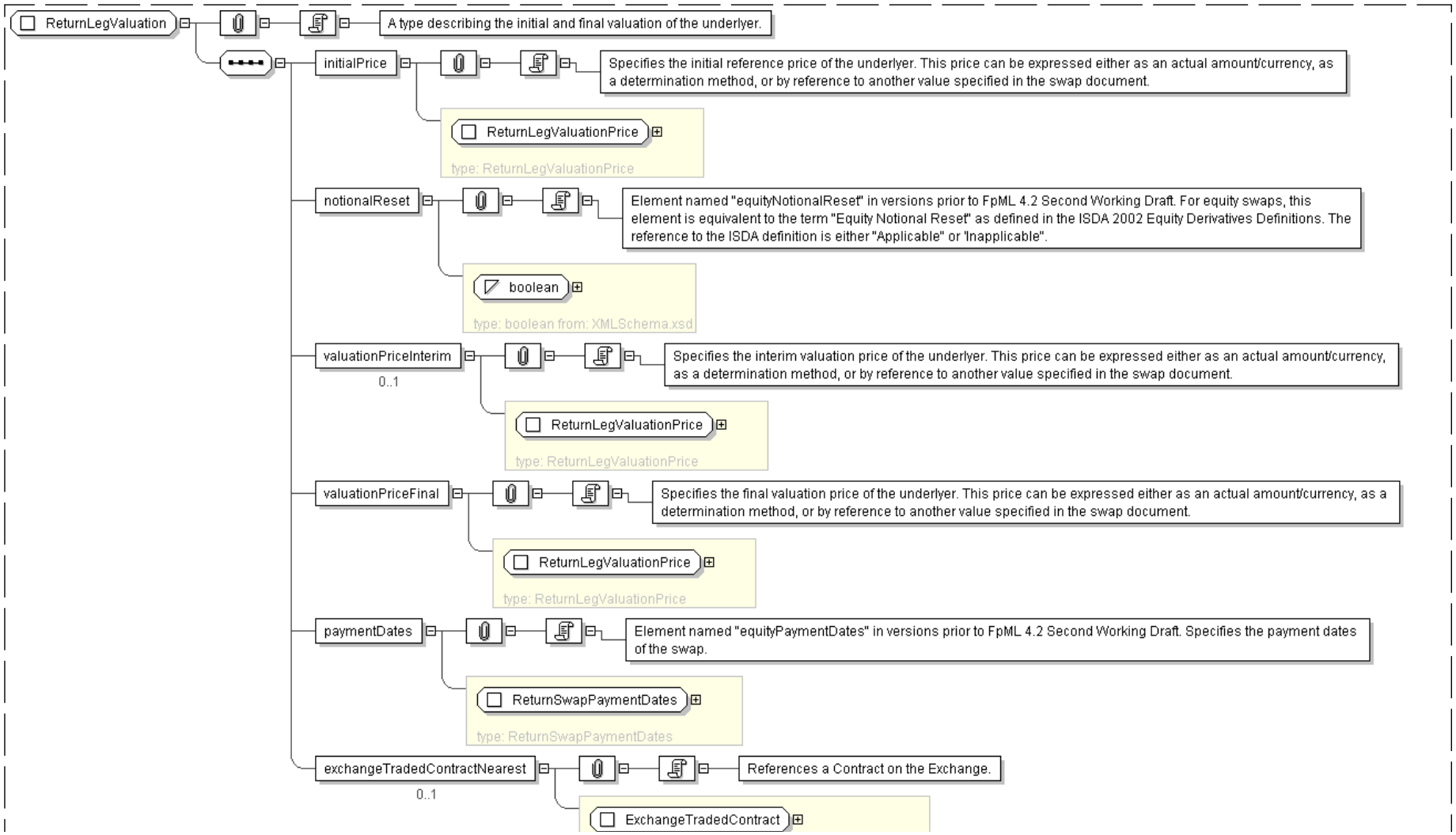
'Element named \"equityPaymentDates\" in versions prior to FpML 4.2 Second Working Draft. Specifies the payment dates of the swap.'

```
<exchangeTradedContractNearest> ExchangeTradedContract </exchangeTradedContractNearest> [0..1]
```

'References a Contract on the Exchange.'

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

Diagram



type: ExchangeTradedContract from: fpml-asset-4-4.xsd

Schema Component Representation

```
<xsd:complexType name="ReturnLegValuation">
   <xsd:sequence>
      <xsd:element name="initialPrice" type=" ReturnLegValuationPrice "/>
      <xsd:element name="notionalReset" type=" xsd:boolean "/>
      <xsd:element name="valuationPriceInterim" type=" ReturnLegValuationPrice " minOccurs="0"/>
      <xsd:element name="valuationPriceFinal" type=" ReturnLegValuationPrice "/>
      <xsd:element name="paymentDates" type=" ReturnSwapPaymentDates "/>
      <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract "
      minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: ReturnLegValuationPrice

Super-types:	Price < **ReturnLegValuationPrice** (by extension)
Sub-types:	None

Name	ReturnLegValuationPrice
Used by (from the same schema document)	Complex Type **ReturnLegValuation** , Complex Type **ReturnLegValuation** , Complex Type **ReturnLegValuation**
Abstract	no

XML Instance Representation

```
<...>
   <commission> Commission </commission> [0..1]
   'This optional component specifies the commission to be charged for executing the
   hedge transactions.'

Start Choice [1]
      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'The href attribute value will be a pointer style reference to the element or
      component elsewhere in the document where the anchor amount is defined.'

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

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

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

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

End Choice
   <cleanNetPrice> xsd:decimal </cleanNetPrice> [0..1]
   'The net price excluding accrued interest. The \"Dirty Price\" for bonds is put in
   the \"netPrice\" element, which includes accrued interest. Thus netPrice - cleanNetPrice
```

```
    = accruedInterest. The currency and price expression for this field are the same as those
    for the (dirty) netPrice.'

    <quotationCharacteristics> QuotationCharacteristics </quotationCharacteristics> [0..1]
    'Allows information about how the price was quoted to be provided.'

    <valuationRules> EquityValuation </valuationRules> [0..1]
    'Element named \"equityValuation\" in versions prior to FpML 4.2 Second Working Draft.'

</...>
```

Diagram

Schema Component Representation

```
<xsd:complexType name="ReturnLegValuationPrice">
   <xsd:complexContent>
      <xsd:extension base=" Price ">
         <xsd:sequence>
            <xsd:element name="valuationRules" type=" EquityValuation " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwap

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

Name	ReturnSwap
Used by (from the same schema document)	Element **returnSwap**
Abstract	no
Documentation	A type describing return swaps including equity swaps (long form), total return swaps, and variance swaps.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
    <productType> ProductType </productType> [0..*]
    'A classification of the type of product. FpML defines a simple product categorization using
    a coding scheme.'

    <productId> ProductId </productId> [0..*]
    'A product reference identifier allocated by a party. FpML does not define the domain
    values associated with this element. Note that the domain values for this element are
    not strictly an enumerated list.'

Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support
the situation where an implementor wishes to indicate who has manufactured the Swap
through representing them as the Seller. It may be removed in future major revisions.'
```

<buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]

'A reference to the party that buys this instrument, ie. pays for this instrument and receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case of FRAs this the fixed rate payer.'

<sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]

'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the rights defined by this instrument and in return receives a payment for it. See 2000 ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

End Group: BuyerSeller.model

<returnSwapLeg> ... </returnSwapLeg> [1..*]

<principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]

'This is used to document a Fully Funded Return Swap.'

<additionalPayment> ReturnSwapAdditionalPayment </additionalPayment> [0..*]

'Specifies additional payment(s) between the principal parties to the trade. This component extends some of the features of the additionalPayment component developed by the FpML industry group. Appropriate discussions will determine whether it would be appropriate to extend the shared component in order to meet the further requirements of equity swaps.'

<earlyTermination> ReturnSwapEarlyTermination </earlyTermination> [0..*]

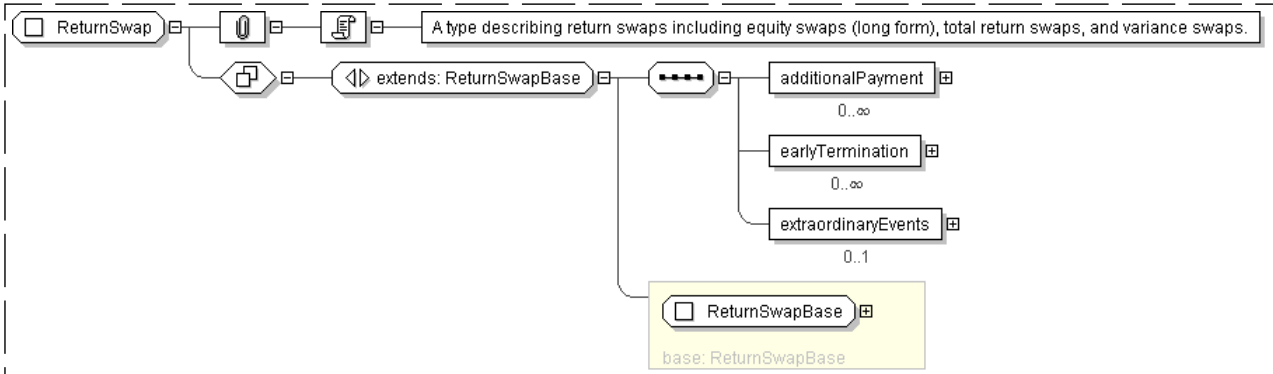
'Specifies, for one or for both the parties to the trade, the date from which it can early terminate it.'

<extraordinaryEvents> ExtraordinaryEvents </extraordinaryEvents> [0..1]

'Where the underlying is shares, specifies events affecting the issuer of those shares that may require the terms of the transaction to be adjusted.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwap">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapBase ">
         <xsd:sequence>
            <xsd:element name="additionalPayment" type=" ReturnSwapAdditionalPayment "
            minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="earlyTermination" type=" ReturnSwapEarlyTermination "
            minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="extraordinaryEvents" type=" ExtraordinaryEvents " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwapAdditionalPayment

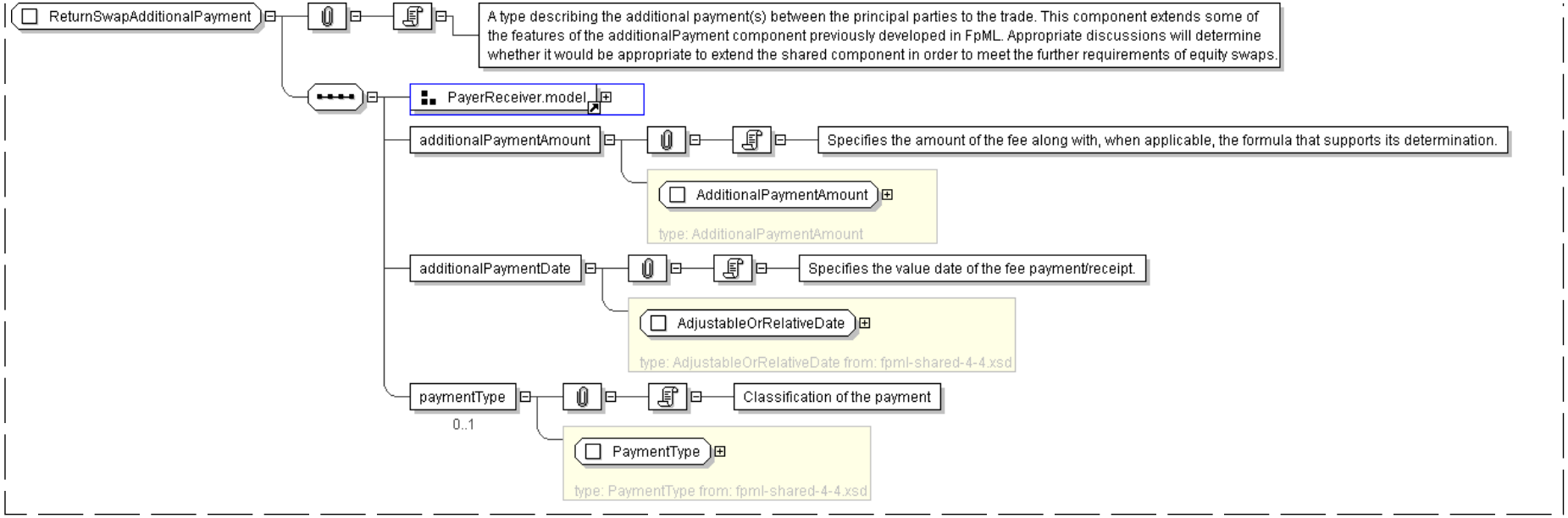
Super-types:	None
Sub-types:	None

Name	ReturnSwapAdditionalPayment
Used by (from the same schema document)	Complex Type **ReturnSwap**
Abstract	no
Documentation	A type describing the additional payment(s) between the principal parties to the trade. This component extends some of the features of the additionalPayment component previously developed in FpML. Appropriate discussions will determine whether it would be appropriate to extend the shared component in order to meet the further requirements of equity swaps.

XML Instance Representation

```
<...>
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <additionalPaymentAmount> AdditionalPaymentAmount </additionalPaymentAmount> [1]
   'Specifies the amount of the fee along with, when applicable, the formula that supports
   its determination.'

   <additionalPaymentDate> AdjustableOrRelativeDate </additionalPaymentDate> [1]
   'Specifies the value date of the fee payment/receipt.'

   <paymentType> PaymentType </paymentType> [0..1]
   'Classification of the payment'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapAdditionalPayment">
   <xsd:sequence>
      <xsd:group ref=" PayerReceiver.model "/>
      <xsd:element name="additionalPaymentAmount" type=" AdditionalPaymentAmount "/>
      <xsd:element name="additionalPaymentDate" type=" AdjustableOrRelativeDate "/>
      <xsd:element name="paymentType" type=" PaymentType " minOccurs="0"/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: ReturnSwapAmount

Super-types:	LegAmount < **ReturnSwapAmount** (by extension)
Sub-types:	• DeprecatedVarianceAmount (by extension)

Name	ReturnSwapAmount
Used by (from the same schema document)	Complex Type **ReturnLeg**
Abstract	no
Documentation	Specifies, in relation to each Payment Date, the amount to which the Payment Date relates. For Equity Swaps this element is equivalent to the Equity Amount term as defined in the ISDA 2002 Equity Derivatives Definitions.

XML Instance Representation

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

      <determinationMethod> DeterminationMethod </determinationMethod> [1]
      'Specifies the method according to which an amount or a date is determined.'

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

End Choice

<paymentCurrency> PaymentCurrency </paymentCurrency> [0..1]

'Currency in which the payment relating to the leg amount (equity amount or interest amount) or the dividend will be denominated.'

Start Choice [1]

<referenceAmount> ReferenceAmount </referenceAmount> [1]

'Specifies the reference Amount when this term either corresponds to the standard ISDA Definition (either the 2002 Equity Definition for the Equity Amount, or the 2000 Definition for the Interest Amount), or points to a term defined elsewhere in the swap document.'

<formula> Formula </formula> [1]

'Specifies a formula, with its description and components.'

<encodedDescription> xsd:base64Binary </encodedDescription> [1]

'Description of the leg amount when represented through an encoded image.'

<variance> DeprecatedVariance </variance> [1]

'DEPRECATED This element will be removed in the next FpML major version. Return Swap model should not be used for Variance Swaps, use the Variance Swap Product. Specifies Variance for Variance Leg.'

End Choice

<calculationDates> AdjustableRelativeOrPeriodicDates </calculationDates> [0..1]

'Specifies the date on which a calculation or an observation will be performed for the purpose of defining the Equity Amount, and in accordance to the definition terms of this latter.'

<cashSettlement> xsd:boolean </cashSettlement> [1]

'If true, then cash settlement is applicable.'

<optionsExchangeDividends> xsd:boolean </optionsExchangeDividends> [0..1]

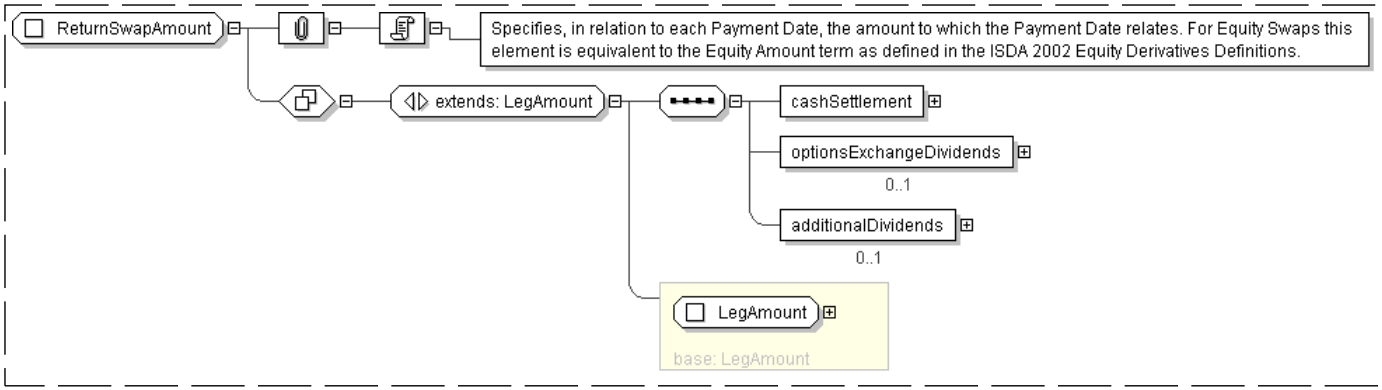
'If present and true, then options exchange dividends are applicable.'

<additionalDividends> xsd:boolean </additionalDividends> [0..1]

'If present and true, then additional dividends are applicable.'

</...>

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapAmount">
   <xsd:complexContent>
      <xsd:extension base=" LegAmount ">
```

```
        <xsd:sequence>
           <xsd:element name="cashSettlement" type=" xsd:boolean "/>
           <xsd:element name="optionsExchangeDividends" type=" xsd:boolean " minOccurs="0"/>
           <xsd:element name="additionalDividends" type=" xsd:boolean " minOccurs="0"/>
        </xsd:sequence>
     </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwapBase

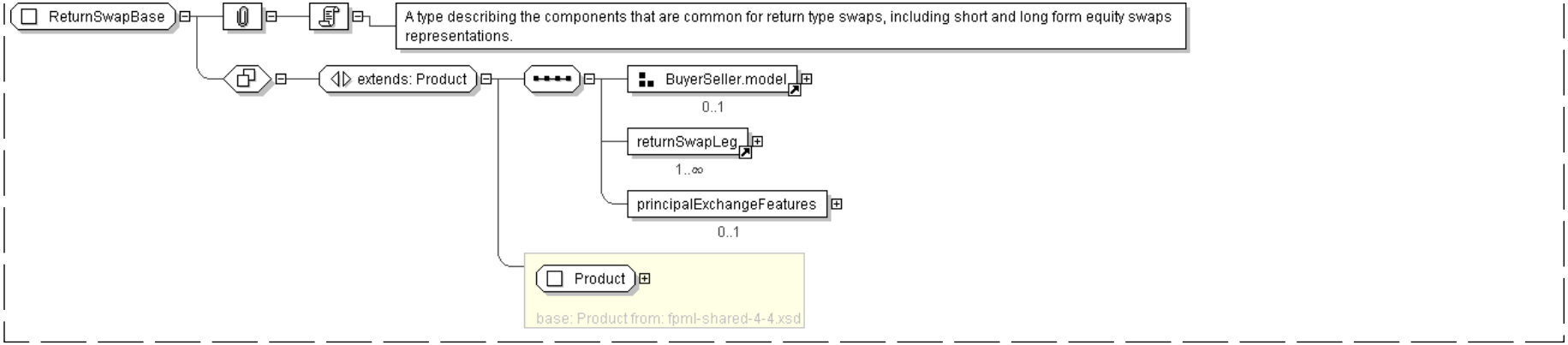
Super-types:	Product < **ReturnSwapBase** (by extension)
Sub-types:	• ReturnSwap (by extension)

Name	ReturnSwapBase
Abstract	yes
Documentation	A type describing the components that are common for return type swaps, including short and long form equity swaps representations.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <productType> ProductType </productType> [0..*]
   'A classification of the type of product. FpML defines a simple product categorization using
   a coding scheme.'

   <productId> ProductId </productId> [0..*]
   'A product reference identifier allocated by a party. FpML does not define the domain
   values associated with this element. Note that the domain values for this element are
   not strictly an enumerated list.'

Start Group: BuyerSeller.model [0..1]
'BuyerSeller.model has been included as an optional child of ReturnSwapBase to support
the situation where an implementor wishes to indicate who has manufactured the Swap
through representing them as the Seller. It may be removed in future major revisions.'


      <buyerPartyReference> PartyOrTradeSideReference </buyerPartyReference> [1]
      'A reference to the party that buys this instrument, ie. pays for this instrument and
      receives the rights defined by it. See 2000 ISDA definitions Article 11.1 (b). In the case
      of FRAs this the fixed rate payer.'

      <sellerPartyReference> PartyOrTradeSideReference </sellerPartyReference> [1]
      'A reference to the party that sells (\"writes\") this instrument, i.e. that grants the
      rights defined by this instrument and in return receives a payment for it. See 2000
      ISDA definitions Article 11.1 (a). In the case of FRAs this is the floating rate payer.'

End Group: BuyerSeller.model
   <returnSwapLeg> ... </returnSwapLeg> [1..*]
   <principalExchangeFeatures> PrincipalExchangeFeatures </principalExchangeFeatures> [0..1]
   'This is used to document a Fully Funded Return Swap.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapBase" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Product ">
         <xsd:sequence>
            <xsd:group ref=" BuyerSeller.model " minOccurs="0"/>
            <xsd:element ref=" returnSwapLeg " maxOccurs="unbounded"/>
            <xsd:element name="principalExchangeFeatures" type=" PrincipalExchangeFeatures " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwapEarlyTermination

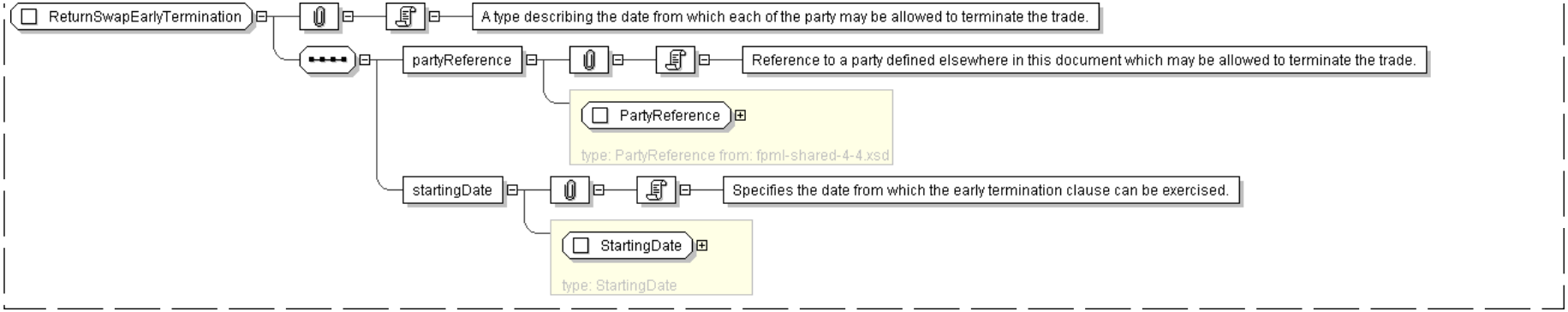
Super-types:	None
Sub-types:	None

Name	ReturnSwapEarlyTermination
Used by (from the same schema document)	Complex Type **ReturnSwap**
Abstract	no
Documentation	A type describing the date from which each of the party may be allowed to terminate the trade.

XML Instance Representation

```
<...>
   <partyReference> PartyReference </partyReference> [1]
   'Reference to a party defined elsewhere in this document which may be allowed to terminate
   the trade.'

   <startingDate> StartingDate </startingDate> [1]
   'Specifies the date from which the early termination clause can be exercised.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapEarlyTermination">
   <xsd:sequence>
      <xsd:element name="partyReference" type=" PartyReference "/>
      <xsd:element name="startingDate" type=" StartingDate "/>
   </xsd:sequence>
</xsd:complexType>
```

top

Complex Type: ReturnSwapLeg

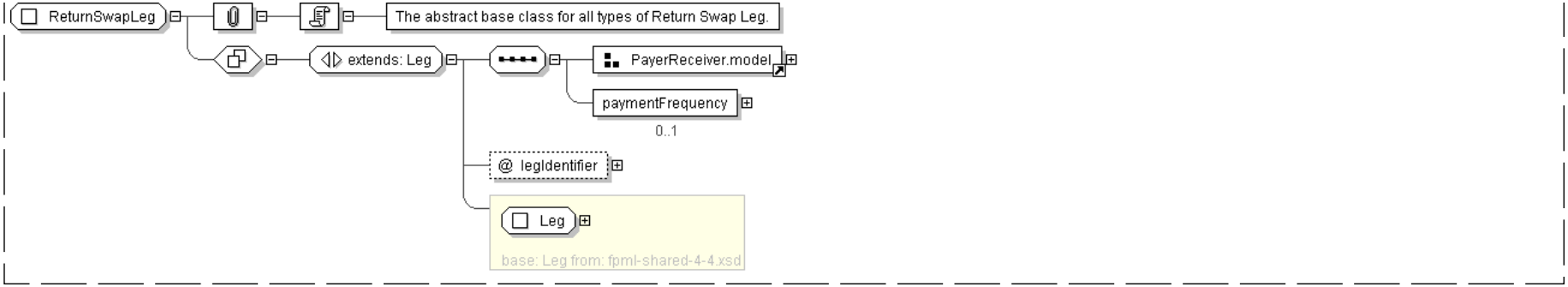
Super-types:	Leg < **ReturnSwapLeg** (by extension)
Sub-types:	• DeprecatedVarianceLeg (by extension) • InterestLeg (by extension) • ReturnSwapLegUnderlyer (by extension) ○ ReturnLeg (by extension)

Name	ReturnSwapLeg
Used by (from the same schema document)	Element **returnSwapLeg**
Abstract	yes
Documentation	The abstract base class for all types of Return Swap Leg.

XML Instance Representation

```
<...
legIdentifier=" xsd:ID [0..1]
 'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapLeg" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" Leg ">
         <xsd:sequence>
            <xsd:group ref=" PayerReceiver.model "/>
            <xsd:element name="paymentFrequency" type=" Interval " minOccurs="0"
            deprecated="true" deprecatedReason="Payment Frequency is controlled by the implementations
            of this abstract base class"/>
         </xsd:sequence>
         <xsd:attribute name="legIdentifier" type=" xsd:ID " deprecated="true"
         deprecatedReason="All attributes of type=xsd:ID should have name=id following
         FpML Architecture"/>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwapLegUnderlyer

Super-types:	Leg < ReturnSwapLeg (by extension) < **ReturnSwapLegUnderlyer** (by extension)
Sub-types:	• ReturnLeg (by extension)

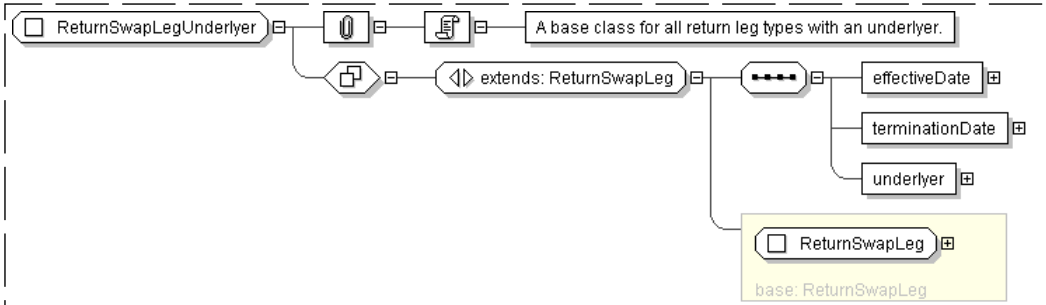
Name	ReturnSwapLegUnderlyer
Abstract	yes
Documentation	A base class for all return leg types with an underlyer.

XML Instance Representation

```
<...
legIdentifier=" xsd:ID [0..1]
'DEPRECATED This element will be renamed to id in the next major FpML version.'

">
   <payerPartyReference> PartyOrAccountReference </payerPartyReference> [1]
   'A reference to the party responsible for making the payments defined by this structure.'

   <receiverPartyReference> PartyOrAccountReference </receiverPartyReference> [1]
   'A reference to the party that receives the payments corresponding to this structure.'

   <paymentFrequency> Interval </paymentFrequency> [0..1]
   'DEPRECATED This element will be removed in the next FpML major version. Frequency at
   which this leg pays.'

   <effectiveDate> AdjustableOrRelativeDate </effectiveDate> [1]
   'Specifies the effective date of this leg of the swap. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component),
```

```
    this element will typically point to the effective date of the other leg of the swap.'

    <terminationDate> AdjustableOrRelativeDate </terminationDate> [1]
    'Specifies the termination date of this leg of the swap. When defined in relation to a
    date specified somewhere else in the document (through the relativeDate component),
    this element will typically point to the termination date of the other leg of the swap.'

    <underlyer> Underlyer </underlyer> [1]
    'Specifies the underlying component of the leg, which can be either one or many and consists
    in either equity, index or convertible bond component, or a combination of these.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapLegUnderlyer" abstract="true">
   <xsd:complexContent>
      <xsd:extension base=" ReturnSwapLeg ">
         <xsd:sequence>
            <xsd:element name="effectiveDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="terminationDate" type=" AdjustableOrRelativeDate "/>
            <xsd:element name="underlyer" type=" Underlyer "/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

Complex Type: ReturnSwapNotional

Super-types:	None
Sub-types:	None

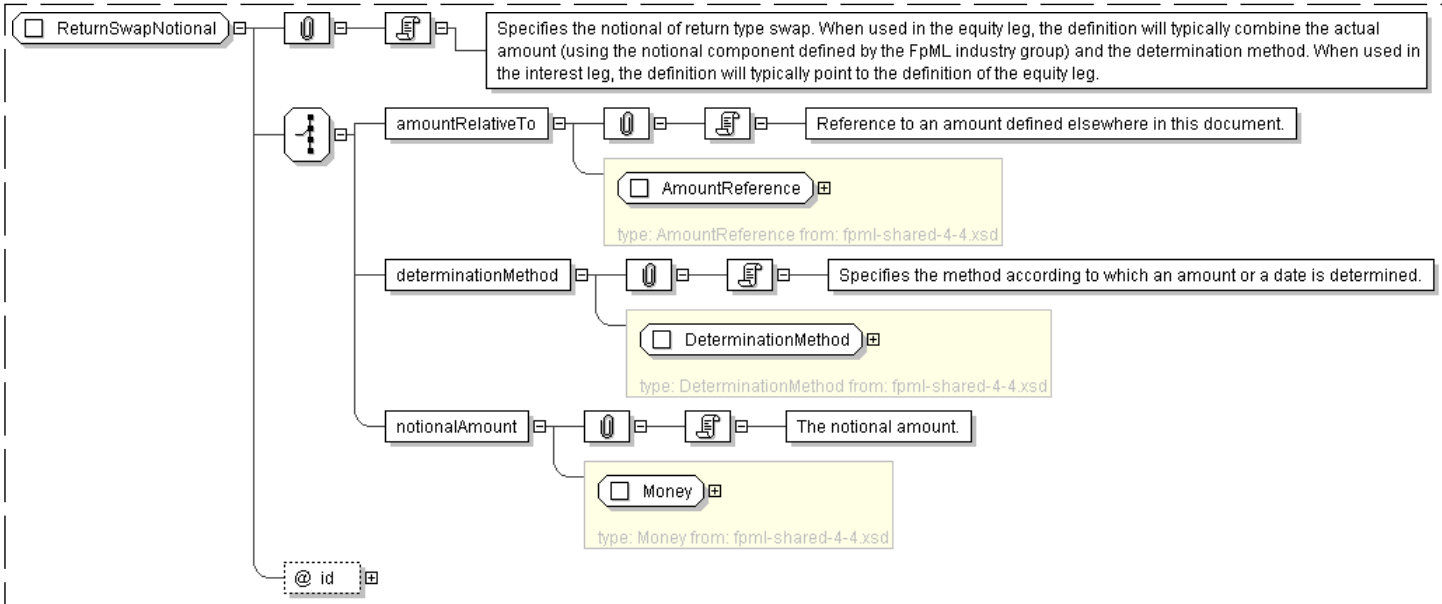
Name	ReturnSwapNotional
Used by (from the same schema document)	Complex Type **InterestLeg** , Complex Type **ReturnLeg**
Abstract	no
Documentation	Specifies the notional of return type swap. When used in the equity leg, the definition will typically combine the actual amount (using the notional component defined by the FpML industry group) and the determination method. When used in the interest leg, the definition will typically point to the definition of the equity leg.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
Start Choice [1]
      <amountRelativeTo> AmountReference </amountRelativeTo> [1]
      'Reference to an amount defined elsewhere in this document.'
```

```
<determinationMethod> DeterminationMethod </determinationMethod> [1]
'Specifies the method according to which an amount or a date is determined.'

<notionalAmount> Money </notionalAmount> [1]
'The notional amount.'

End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapNotional">
   <xsd:choice>
      <xsd:element name="amountRelativeTo" type=" AmountReference "/>
      <xsd:element name="determinationMethod" type=" DeterminationMethod "/>
      <xsd:element name="notionalAmount" type=" Money "/>
   </xsd:choice>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

top

Complex Type: ReturnSwapPaymentDates

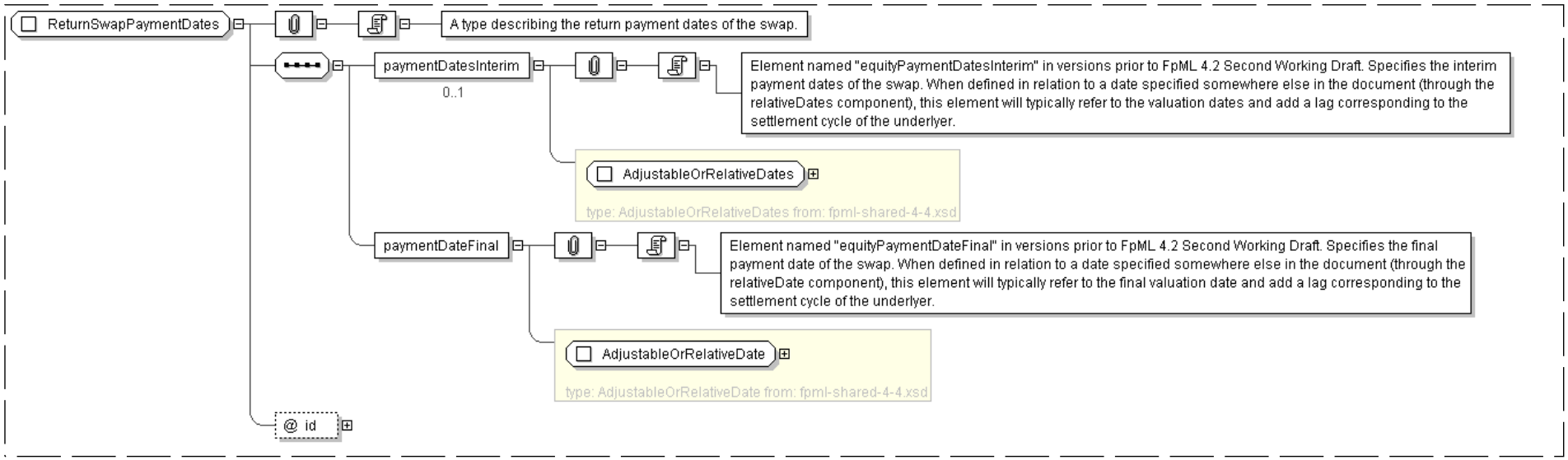
Super-types:	None
Sub-types:	None

Name	ReturnSwapPaymentDates
Used by (from the same schema document)	Complex Type **ReturnLegValuation**
Abstract	no
Documentation	A type describing the return payment dates of the swap.

XML Instance Representation

```
<...
id=" xsd:ID [0..1]">
   <paymentDatesInterim> AdjustableOrRelativeDates </paymentDatesInterim> [0..1]
   'Element named \"equityPaymentDatesInterim\" in versions prior to FpML 4.2 Second
   Working Draft. Specifies the interim payment dates of the swap. When defined in relation to
   a date specified somewhere else in the document (through the relativeDates component),
   this element will typically refer to the valuation dates and add a lag corresponding to
   the settlement cycle of the underlyer.'

   <paymentDateFinal> AdjustableOrRelativeDate </paymentDateFinal> [1]
   'Element named \"equityPaymentDateFinal\" in versions prior to FpML 4.2 Second Working
   Draft. Specifies the final payment date of the swap. When defined in relation to a
   date specified somewhere else in the document (through the relativeDate component),
   this element will typically refer to the final valuation date and add a lag corresponding
   to the settlement cycle of the underlyer.'

</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="ReturnSwapPaymentDates">
   <xsd:sequence>
      <xsd:element name="paymentDatesInterim" type=" AdjustableOrRelativeDates " minOccurs="0"/>
      <xsd:element name="paymentDateFinal" type=" AdjustableOrRelativeDate "/>
   </xsd:sequence>
   <xsd:attribute name="id" type=" xsd:ID "/>
</xsd:complexType>
```

top

Complex Type: StartingDate

Super-types:	None
Sub-types:	None

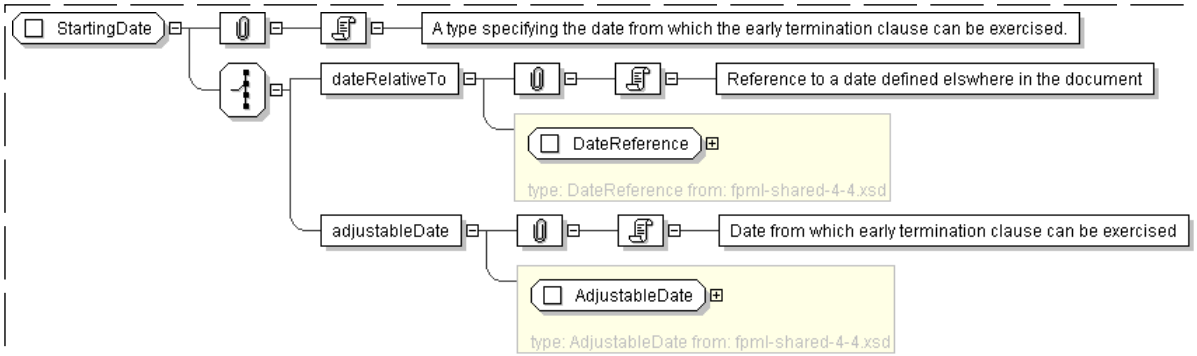
Name	StartingDate
Used by (from the same schema document)	Complex Type **DeprecatedVarianceAmount** , Complex Type **ReturnSwapEarlyTermination**
Abstract	no

Documentation	A type specifying the date from which the early termination clause can be exercised.

XML Instance Representation

```
<...>
Start Choice [1]
      <dateRelativeTo> DateReference </dateRelativeTo> [1]
      'Reference to a date defined elswhere in the document'

      <adjustableDate> AdjustableDate </adjustableDate> [1]
      'Date from which early termination clause can be exercised'

End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StartingDate">
   <xsd:choice>
      <xsd:element name="dateRelativeTo" type=" DateReference "/>
      <xsd:element name="adjustableDate" type=" AdjustableDate "/>
   </xsd:choice>
</xsd:complexType>
```

top

Complex Type: StubCalculationPeriod

Super-types:	None
Sub-types:	None

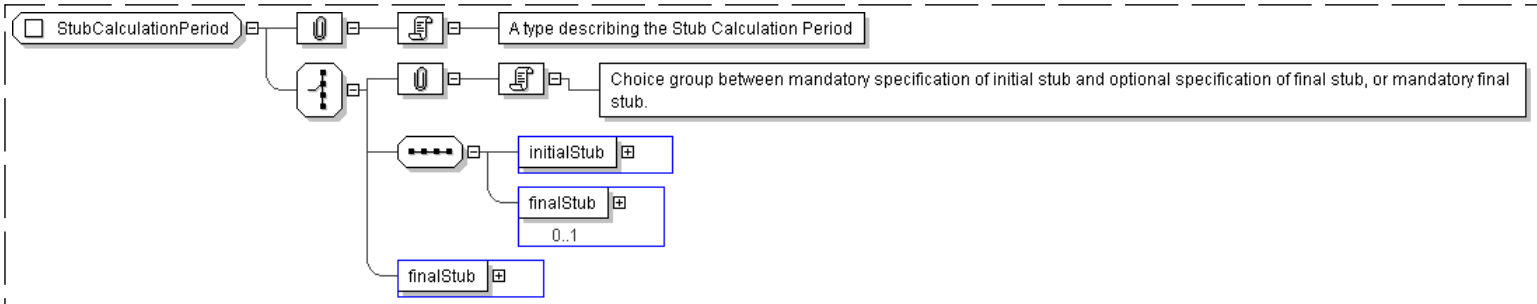
Name	StubCalculationPeriod
Used by (from the same schema document)	Complex Type **InterestLeg**
Abstract	no
Documentation	A type describing the Stub Calculation Period

XML Instance Representation

```
<...>
Start Choice [1]
'Choice group between mandatory specification of initial stub and optional specification
of final stub, or mandatory final stub.'

      <initialStub> Stub </initialStub> [1]
      <finalStub> Stub </finalStub> [0..1]
```

```
      <finalStub> Stub </finalStub> [1]
End Choice
</...>
```

Diagram



Schema Component Representation

```
<xsd:complexType name="StubCalculationPeriod">
   <xsd:choice>
      <xsd:sequence>
         <xsd:element name="initialStub" type=" Stub "/>
         <xsd:element name="finalStub" type=" Stub " minOccurs="0"/>
      </xsd:sequence>
      <xsd:element name="finalStub" type=" Stub "/>
   </xsd:choice>
</xsd:complexType>
```

top

Complex Type: Variance

Super-types:	CalculationFromObservation < **Variance** (by extension)
Sub-types:	None

Name	Variance
Abstract	no
Documentation	A type describing the variance amount of a variance swap

XML Instance Representation

```
<...>
Start Choice [1]
      <initialLevel> xsd:decimal </initialLevel> [1]
      'Contract will strike off this initial level'

      <closingLevel> xsd:boolean </closingLevel> [1]
      'If true this contract will strike off the closing level of the default exchange
      traded contract'

      <expiringLevel> xsd:boolean </expiringLevel> [1]
      'If true this contract will strike off the expiring level of the default exchange
      traded contract'

End Choice
   <expectedN> xsd:positiveInteger </expectedN> [0..1]
   'Expected number of trading days'

   <varianceAmount> Money </varianceAmount> [1]
   'Variance amount, which is a cash multiplier.'
```

Start Choice [1]

'Choice between expressing the strike as volatility or variance.'

<volatilityStrikePrice> NonNegativeDecimal **</volatilityStrikePrice>** [1]

<varianceStrikePrice> NonNegativeDecimal **</varianceStrikePrice>** [1]

End Choice

<varianceCap> xsd:boolean **</varianceCap>** [0..1]

'If present and true, then variance cap is applicable.'

<unadjustedVarianceCap> PositiveDecimal **</unadjustedVarianceCap>** [0..1]

'For use when varianceCap is applicable. Contains the scaling factor of the Variance Cap that can differ on a trade-by-trade basis in the European market. For example, a Variance Cap of 2.5^2 x Variance Strike Price has an unadjustedVarianceCap of 2.5.'

<boundedVariance> BoundedVariance **</boundedVariance>** [0..1]

'Conditions which bound variance. The contract specifies one or more boundary levels. These levels are expressed as prices for confirmation purposes Underlyer price must be equal to or higher than Lower Barrier is known as Up Conditional Swap Underlyer price must be equal to or lower than Upper Barrier is known as Down Conditional Swap Underlyer price must be equal to or higher than Lower Barrier and must be equal to or lower than Upper Barrier is known as Barrier Conditional Swap.'

<exchangeTradedContractNearest> ExchangeTradedContract **</exchangeTradedContractNearest>** [0..1]

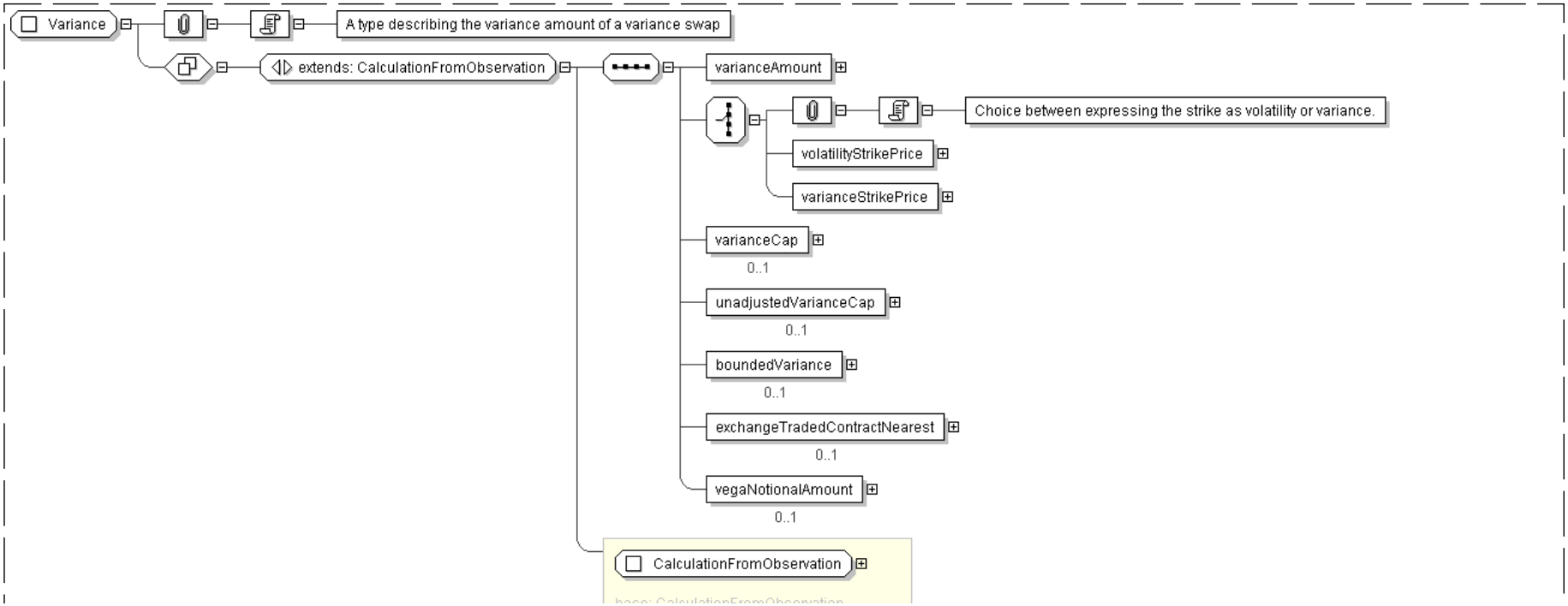
'Specification of the exchange traded contract nearest.'

<vegaNotionalAmount> xsd:decimal **</vegaNotionalAmount>** [0..1]

'Vega Notional represents the approximate gain/loss at maturity for a 1% difference between RVol (realised vol) and KVol (strike vol). It does not necessarily represent the Vega Risk of the trade.'

</...>

Diagram



base: CalculationFromObservation

Schema Component Representation

```
<xsd:complexType name="Variance">
   <xsd:complexContent>
      <xsd:extension base=" CalculationFromObservation ">
         <xsd:sequence>
            <xsd:element name="varianceAmount" type=" Money "/>
            <xsd:choice>
               <xsd:element name="volatilityStrikePrice" type=" NonNegativeDecimal "/>
               <xsd:element name="varianceStrikePrice" type=" NonNegativeDecimal "/>
            </xsd:choice>
            <xsd:element name="varianceCap" type=" xsd:boolean " minOccurs="0"/>
            <xsd:element name="unadjustedVarianceCap" type=" PositiveDecimal " minOccurs="0"/>
            <xsd:element name="boundedVariance" type=" BoundedVariance " minOccurs="0"/>
            <xsd:element name="exchangeTradedContractNearest" type=" ExchangeTradedContract "
            minOccurs="0"/>
            <xsd:element name="vegaNotionalAmount" type=" xsd:decimal " minOccurs="0"/>
         </xsd:sequence>
      </xsd:extension>
   </xsd:complexContent>
</xsd:complexType>
```

top

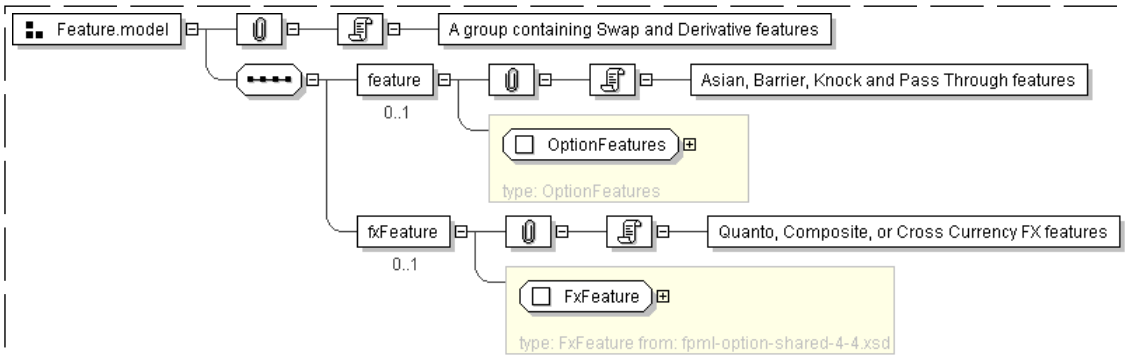
Model Group: Feature.model

Name	Feature.model
Documentation	A group containing Swap and Derivative features

XML Instance Representation

```
<feature> OptionFeatures </feature> [0..1]
'Asian, Barrier, Knock and Pass Through features'

<fxFeature> FxFeature </fxFeature> [0..1]
'Quanto, Composite, or Cross Currency FX features'
```

Diagram



Schema Component Representation

```
<xsd:group name="Feature.model">
   <xsd:sequence>
      <xsd:element name="feature" type=" OptionFeatures " minOccurs="0"/>
      <xsd:element name="fxFeature" type=" FxFeature " minOccurs="0"/>
   </xsd:sequence>
</xsd:group>
```

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.