



FpML Shared Component Definitions

Working Draft 30 January 2002

Version: 3.0

This Version:

<http://www.fpml.org/spec/2002/wd-fpml-shared-componentDefinitions-3-0-2002-01-30>

Latest Version:

<http://www.fpml.org/spec/fpml-3-0>

Previous Version:

<http://www.fpml.org/spec/2002/tr-fpml-2-0-2002-01-14>

Copyright 1999 - 2002. All rights reserved.

Financial Products Markup Language is subject to the FpML Public License.

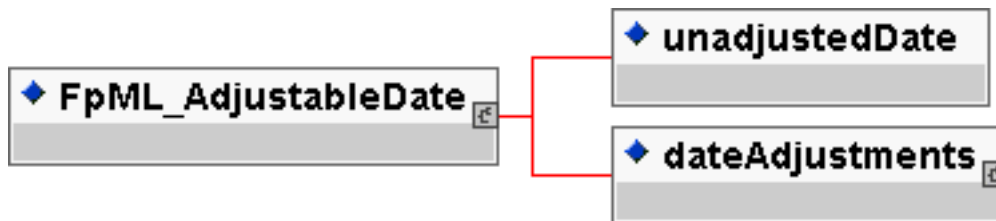
A copy of this license is available at <http://www.fpml.org/documents/license>

FpML_AdjustableDate

Description:

An entity for defining a date that shall be subject to adjustment if it would otherwise fall on a day that is not a business day in the specified business centers, together with the convention for adjusting the date.

Figure:



Contents:

unadjustedDate (exactly one occurrence; of type *date*)

- A date subject to adjustment.

dateAdjustments (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessDayAdjustments)

- The business day convention and financial business centers used for adjusting the date if it would otherwise fall on a day that is not a business day in the specified business centers.

Used by:

- adjustableDate
- effectiveDate
- firstPeriodStartDate
- mandatoryEarlyTerminationDate
- paymentDate
- terminationDate

DTD Fragment:

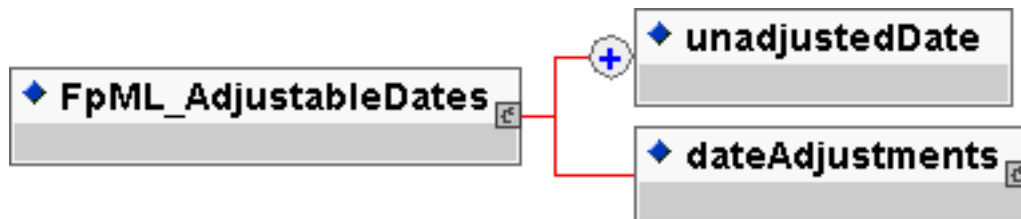
```
<!ENTITY % FpML_AdjustableDate "unadjustedDate , dateAdjustments">
```

FpML_AdjustableDates

Description:

An entity for defining a series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified business centers, together with the convention for adjusting the dates.

Figure:



Contents:

unadjustedDate (one or more occurrences; of type *date*)

- A date subject to adjustment.

dateAdjustments (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessDayAdjustments)

- The business day convention and financial business centers used for adjusting the date if it would otherwise fall on a day that is not a business day in the specified business centers.

Used by:

- adjustableDates

DTD Fragment:

```
<!ENTITY % FpML_AdjustableDates "unadjustedDate+ , dateAdjustments">
```

FpML_AdjustableOrRelativeDate

Description:

An entity for the choice between defining a date as an explicit date together with applicable adjustments or as relative to some other (anchor) date.

Figure:



Contents:

Either

adjustableDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableDate)

- 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.

Or

relativeDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_RelativeDateOffset)

- A date specified as some offset to another date (the anchor date).

Used by:

- commencementDate
- expirationDate

DTD Fragment:

```
<!ENTITY % FpML_AdjustableOrRelativeDate "adjustableDate | relativeDate">
```

FpML_AdjustableOrRelativeDates

Description:

An entity for the choice between defining a series of dates as an explicit list of dates together with applicable adjustments or as relative to some other series of (anchor) dates.

Figure:



Contents:

Either

adjustableDates (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableDates)

- A series of dates that shall be subject to adjustment if they would otherwise fall on a day that is not a business day in the specified business centers, together with the convention for adjusting the date.

Or

relativeDates (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_RelativeDates)

- A series of dates specified as some offset to another series of dates. (the anchor dates).

Used by:

- bermudanExerciseDates
- relevantUnderlyingDate

DTD Fragment:

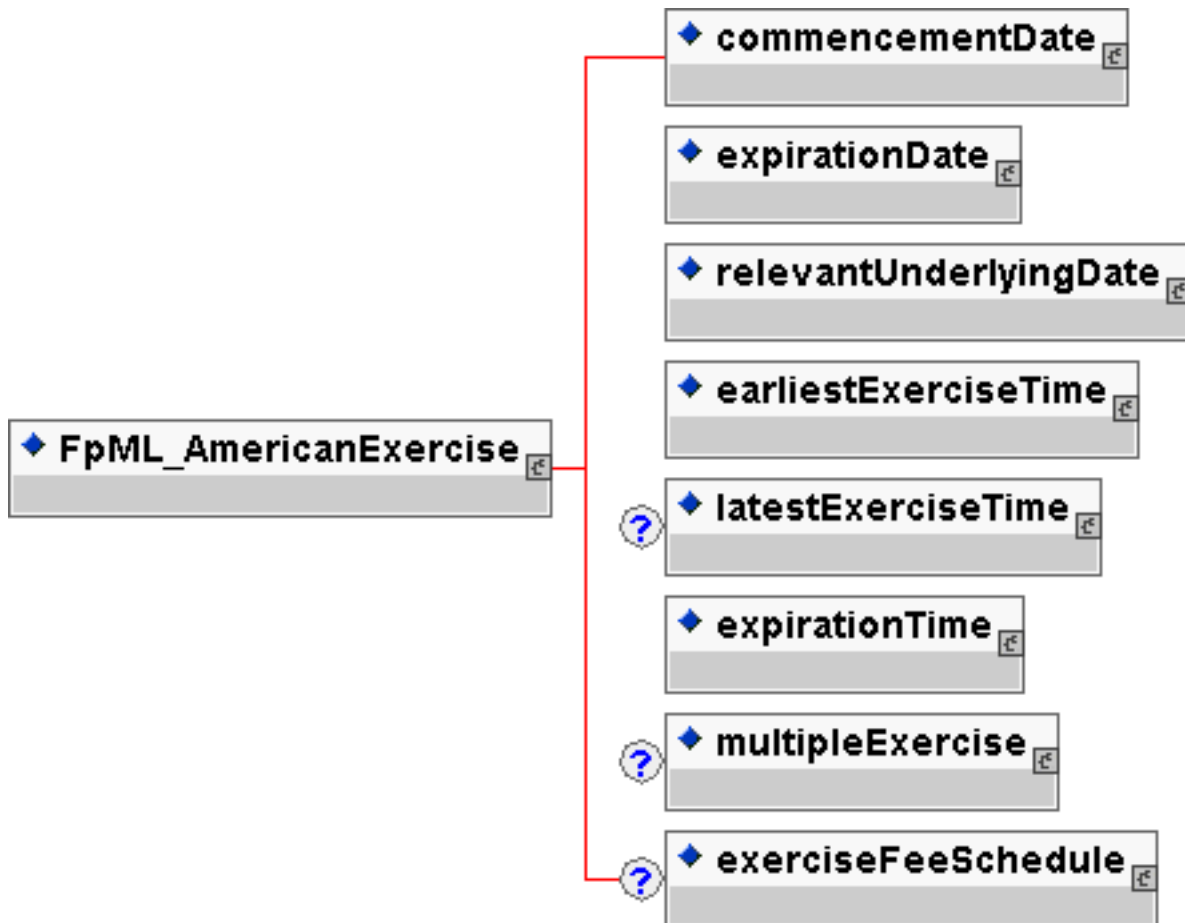
```
<!ENTITY % FpML_AdjustableOrRelativeDates "adjustableDates | relativeDates">
```

FpML_AmericanExercise

Description:

An entity for defining the exercise period for an American style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Figure:



Contents:

commencementDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDate)

- The first day of the exercise period for an American style option.

expirationDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDate)

- The last day within an exercise period for an American style option. For a European style option it is the only day within the exercise period.

relevantUnderlyingDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDates)

- The date on the underlying set by the exercise of an option. What this date is depends on

the option (eg in a swaption it is the effective date, in a extendible / cancelable provision is is the termination date).

earliestExerciseTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The earliest time at which notice of exercise can be given by the buyer to the seller (or seller's agent) i) on the expiration date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in the case of a Bermudan style option and (iii) all days that are exercise business days from and including the commencement date to, and including, the expiration date, in the case of an American style option.

latestExerciseTime (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- For a Bermudan or American style options, the latest time on an exercise business day (excluding the expiration date) within the exercise period that notice of exercise can be given by buyer to the seller or seller's agent. Notice of exercise given after this time will be deemed to have been given on the next exercise business day.

expirationTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The latest time for expiration on expirationDate.

multipleExercise (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_MultipleExercise)

- As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on any such day may not exercise less than the minimum notional amount or more than the maximum notional amount, and if an integral multiple amount is specified, the notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.

exerciseFeeSchedule (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_ExerciseFeeSchedule)

- The fees associated with an exercise date. The fees are conditional on the exercise occurring. The fees can be specified as actual currency amounts or as percentages of the notional amount being exercised.

Used by:

- americanExercise

DTD Fragment:

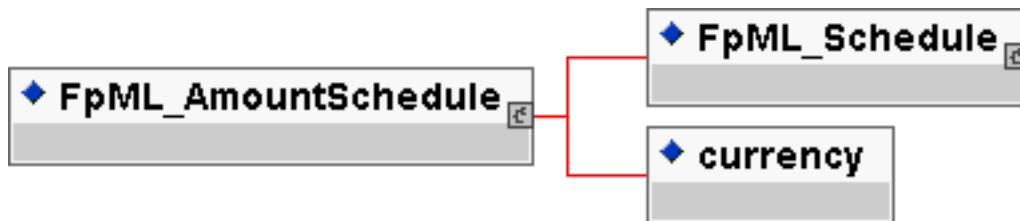
```
<!ENTITY % FpML_AmericanExercise "commencementDate , expirationDate , relevantUnderlyingDate ,
earliestExerciseTime , latestExerciseTime? , expirationTime , multipleExercise? , exerciseFeeSchedule?">
```


FpML_AmountSchedule

Description:

An entity for defining a currency amount or a currency amount schedule. This entity inherits from a base entity, FpML_Schedule.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_Schedule)

- An entity for defining a schedule of rate or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

currency (exactly one occurrence; of type *string*, an enumerated domain value defined by *currencyScheme*)

- The currency in which an amount is denominated.

Used by:

- knownAmountSchedule
- notionalStepSchedule

DTD Fragment:

```
<!ENTITY % FpML_AmountSchedule "(%FpML_Schedule; , currency)">
```

FpML_AutomaticExercise

Description:

An entity to define automatic exercise of a swaption. With automatic exercise the option is deemed to have exercised if it is in the money by more than the threshold amount on the exercise date.

Figure:



Contents:

thresholdRate (exactly one occurrence; of type *decima*)

- A threshold rate. A threshold of 0.10% would be represented as 0.001.

Used by:

- automaticExercise

DTD Fragment:

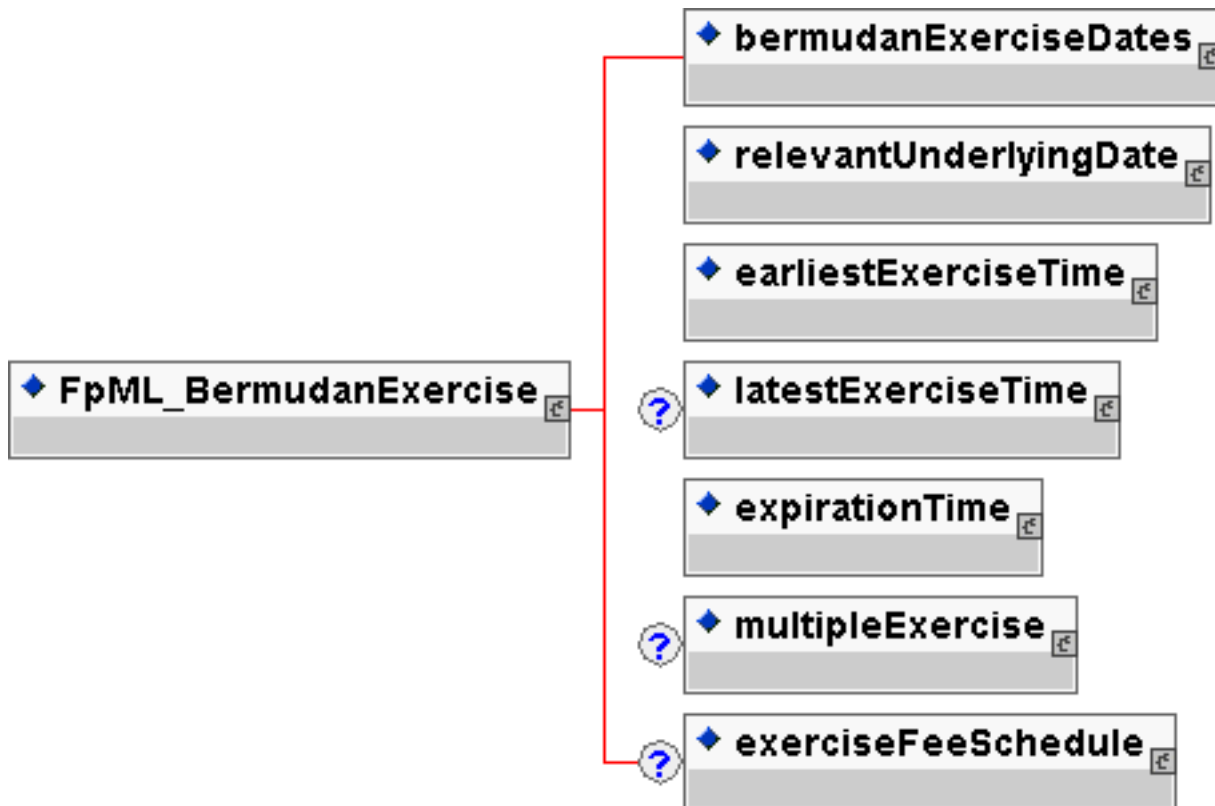
```
<!ENTITY % FpML_AutomaticExercise "thresholdRate">
```

FpML_BermudanExercise

Description:

An entity to define the bermudan option exercise dates and the expiration date together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Figure:



Contents:

bermudanExerciseDates (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDates)

- The dates that define the bermudan option exercise dates and the expiration date. The last specified exercise date is assumed to be the expiration date. The dates can either be specified as a series of explicit dates and associated adjustments or as a series of dates defined relative to another schedule of dates, for example, the calculation period start dates. Where a relative series of dates are defined the first and last possible exercise dates can be separately specified.

relevantUnderlyingDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDates)

- The date on the underlying set by the exercise of an option. What this date is depends on the option (eg in a swaption it is the effective date, in a extendible / cancelable provision is the termination date).

earliestExerciseTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The earliest time at which notice of exercise can be given by the buyer to the seller (or seller's agent) i) on the expiration date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in the case of a Bermudan style option and (iii) all days that are exercise business days from and including the commencement date to, and including, the expiration date, in the case of an American style option.

latestExerciseTime (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- For a Bermudan or American style options, the latest time on an exercise business day (excluding the expiration date) within the exercise period that notice of exercise can be given by buyer to the seller or seller's agent. Notice of exercise given after this time will be deemed to have been given on the next exercise business day.

expirationTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The latest time for expiration on expirationDate.

multipleExercise (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_MultipleExercise)

- As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on any such day may not exercise less than the minimum notional amount or more than the maximum notional amount, and if an integral multiple amount is specified, the notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.

exerciseFeeSchedule (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_ExerciseFeeSchedule)

- The fees associated with an exercise date. The fees are conditional on the exercise occurring. The fees can be specified as actual currency amounts or as percentages of the notional amount being exercised.

Used by:

- bermudanExercise

DTD Fragment:

```
<!ENTITY % FpML_BermudanExercise "bermudanExerciseDates , relevantUnderlyingDate ,
earliestExerciseTime , latestExerciseTime? , expirationTime , multipleExercise? , exerciseFeeSchedule?">
```

FpML_BusinessCenters

Description:

An entity for defining financial business centers used in determining whether a day is a business day or not.

Figure:



Contents:

businessCenter (one or more occurrences; of type *string*, an enumerated domain value defined by *businessCenterScheme*)

- A code identifying a financial business center location. A list of business centers may be ordered in the document alphabetically based on business center code. An FpML document containing an unordered business center list is still regarded as a conformant document.

Used by:

- businessCenters

DTD Fragment:

```
<!ENTITY % FpML_BusinessCenters "businessCenter+">
```

FpML_BusinessCenterTime

Description:

An entity for defining a time with respect to a business center location. For example, 11:00 am London time.

Figure:



Contents:

hourMinuteTime (exactly one occurrence; of type *time*)

- A time specified in hh:mm:ss format where the second component must be '00', e.g. 11am would be represented as 11:00:00.

businessCenter (exactly one occurrence; of type *string*, an enumerated domain value defined by *businessCenterScheme*)

- A code identifying a financial business center location. A list of business centers may be ordered in the document alphabetically based on business center code. An FpML document containing an unordered business center list is still regarded as a conformant document.

Used by:

- FpML_ExpiryDateTime
- cashSettlementValuationTime
- earliestExerciseTime
- expirationTime
- fixingTime
- latestExerciseTime

DTD Fragment:

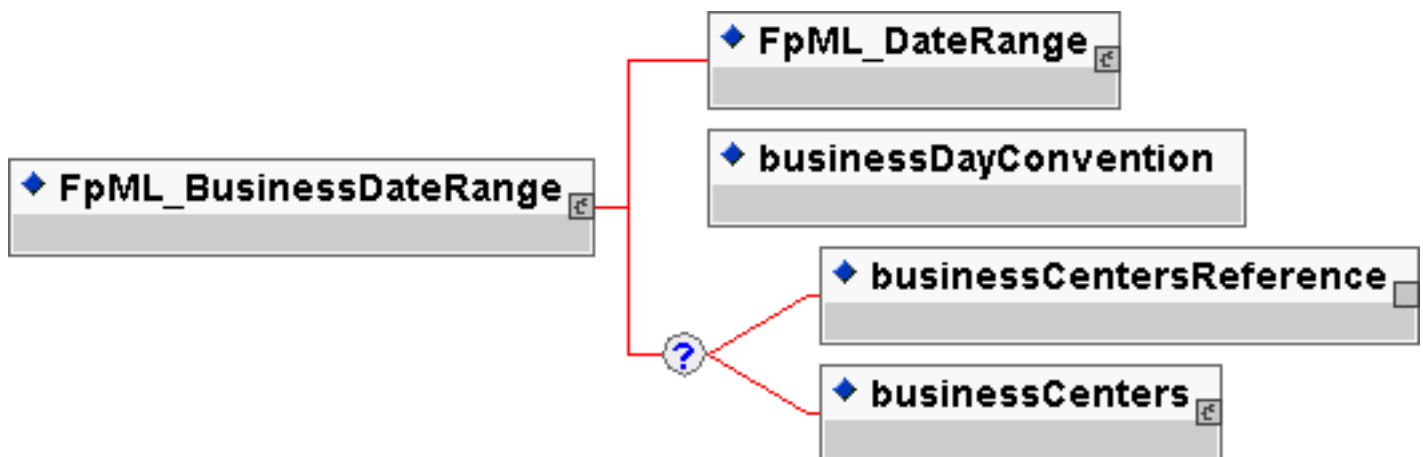
```
<!ENTITY % FpML_BusinessCenterTime "hourMinuteTime , businessCenter">
```

FpML_BusinessDateRange

Description:

An entity for defining a range of contiguous business days by defining an unadjusted first date, an unadjusted last date and a business day convention and business centers for adjusting the first and last dates if they would otherwise fall on a non business day in the specified business centers. The days between the first and last date must also be good business days in the specified business centers to be counted in the range. This entity inherits from the base entity, FpML_DateRange.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_DateRange)

- A entity for defining a contiguous series of calendar dats. The date range is defined as all the dates between and including the first and the last date. The first date must fall before the last date.

businessDayConvention (exactly one occurrence; of type *string*, an enumerated domain value defined by *businessDayConventionScheme*)

- The convention for adjusting a date if it would otherwise fall on a day that is not a business day.

Zero or one occurrence of either

businessCentersReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to determine whether a particular day is a business day or not.

Or

businessCenters (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenters)

- A container for a set of financial business centers. This set of business centers is used to determine whether a day is a business day or not.

Used by:

- businessDateRange

DTD Fragment:

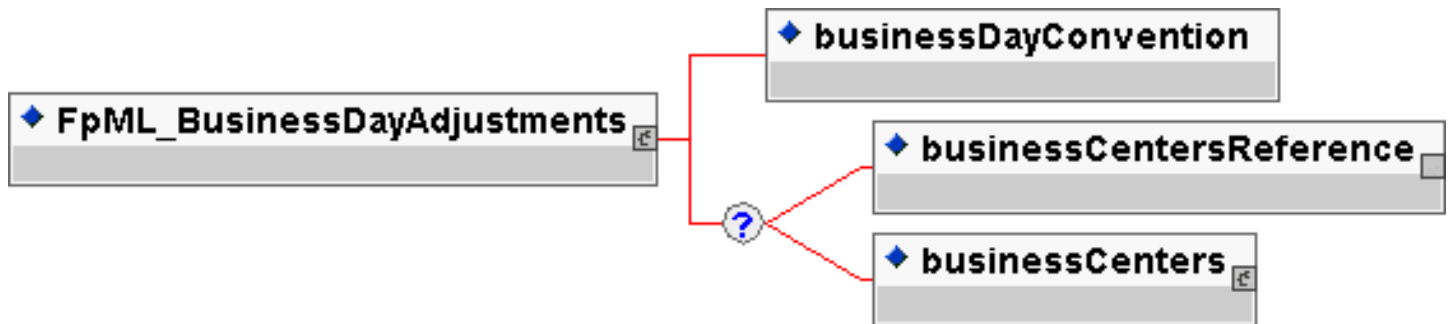
```
<!ENTITY % FpML_BusinessDateRange "(%FpML_DateRange; , businessDayConvention ,  
(businessCentersReference | businessCenters)?)">
```


FpML_BusinessDayAdjustments

Description:

An entity for defining the business day convention and financial business centers used for adjusting any relevant date if it would otherwise fall on a day that is not a business day in the specified business centers.

Figure:



Contents:

businessDayConvention (exactly one occurrence; of type *string*, an enumerated domain value defined by *businessDayConventionScheme*)

- The convention for adjusting a date if it would otherwise fall on a day that is not a business day.

Zero or one occurrence of either

businessCentersReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to determine whether a particular day is a business day or not.

Or

businessCenters (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenters)

- A container for a set of financial business centers. This set of business centers is used to determine whether a day is a business day or not.

Used by:

- calculationPeriodDatesAdjustments
- dateAdjustments
- paymentDatesAdjustments
- resetDatesAdjustments

DTD Fragment:

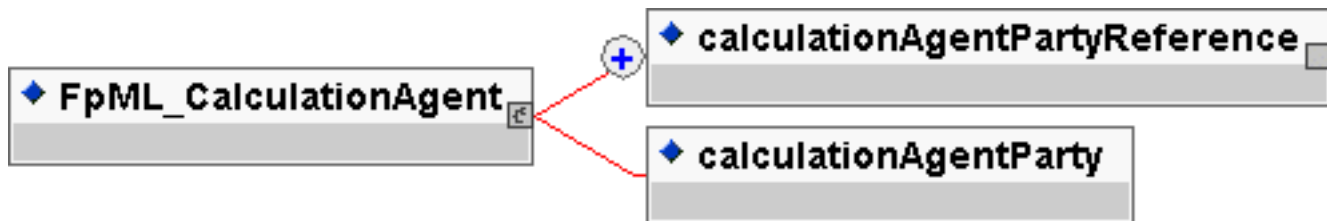
```
<ENTITY % FpML_BusinessDayAdjustments "businessDayConvention , (businessCentersReference | businessCenters)?">
```

FpML_CalculationAgent

Description:

An entity for defining the ISDA Calculation Agent responsible for performing duties associated with the optional early termination on a swap transaction.

Figure:



Contents:

Either

calculationAgentPartyReference (one or more occurrences; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document. The party referenced is the ISDA Calculation Agent for the trade. If more than one party is referenced then the parties are assumed to be co-calculation agents, i.e. they have joint responsibility.

Or

calculationAgentParty (exactly one occurrence; of type *string*, an enumerated domain value defined by *calculationAgentPartyScheme*)

- The ISDA Calculation Agent where the actual party responsible for performing the duties associated with an optional early termination provision will be determined at exercise. For example, the Calculation Agent may be defined as being the Non-exercising Party.

Used by:

- calculationAgent

DTD Fragment:

```
<!ENTITY % FpML_CalculationAgent "calculationAgentPartyReference+ | calculationAgentParty">
```

FpML_CalculationPeriodFrequency

Description:

An entity for defining the frequency at which calculation period end dates occur within the regular part of the calculation period schedule and their roll date convention. This entity inherits from a base entity, FpML_Interval.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_Interval)

- An entity for defining a time interval or offset, e.g. one day, three months. Used for specifying frequencies at which events occur, the tenor of a floating rate or an offset relative to another date.

rollConvention (exactly one occurrence; of type *string*, an enumerated domain value defined by *rollConventionScheme*)

- Used in conjunction with a frequency and the regular period start date of a calculation period, determines each calculation period end date within the regular part of a calculation period schedule.

Used by:

- calculationPeriodFrequency

DTD Fragment:

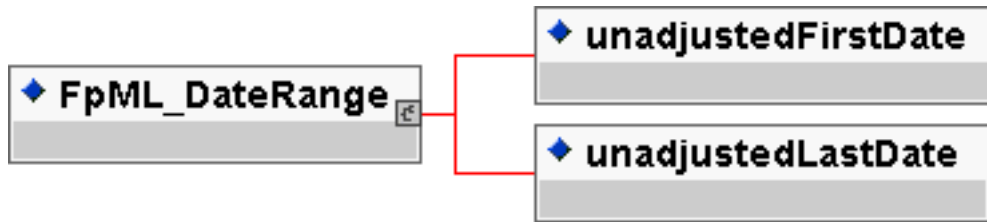
```
<!ENTITY % FpML_CalculationPeriodFrequency "(%FpML_Interval; , rollConvention)">
```

FpML_DateRange

Description:

A entity for defining a contiguous series of calendar dats. The date range is defined as all the dates between and including the first and the last date. The first date must fall before the last date.

Figure:



Contents:

unadjustedFirstDate (exactly one occurrence; of type *date*)

- The first date of a date range.

unadjustedLastDate (exactly one occurrence; of type *date*)

- The last date of a date range.

Used by:

- FpML_BusinessDateRange
- scheduleBounds

DTD Fragment:

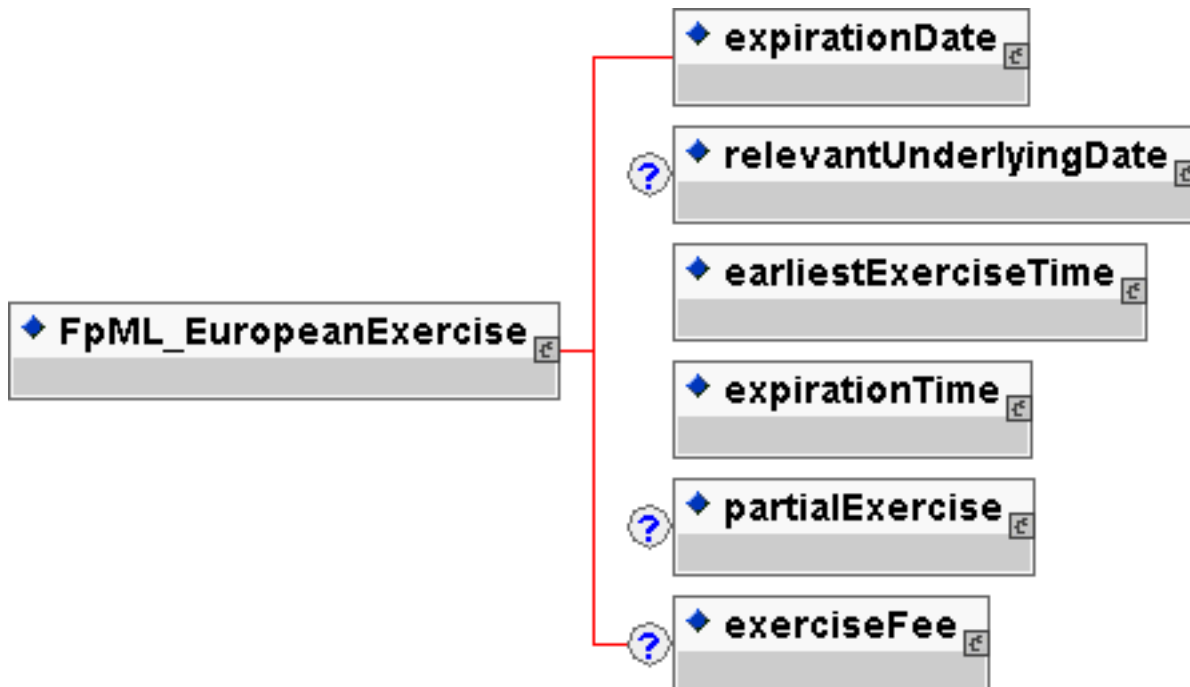
```
<!ENTITY % FpML_DateRange "unadjustedFirstDate , unadjustedLastDate">
```

FpML_EuropeanExercise

Description:

An entity to define the exercise period for a European style option together with any rules governing the notional amount of the underlying which can be exercised on any given exercise date and any associated exercise fees.

Figure:



Contents:

expirationDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDate)

- The last day within an exercise period for an American style option. For a European style option it is the only day within the exercise period.

relevantUnderlyingDate (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableOrRelativeDates)

- The date on the underlying set by the exercise of an option. What this date is depends on the option (eg in a swaption it is the effective date, in a extendible / cancelable provision is the termination date).

earliestExerciseTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The earliest time at which notice of exercise can be given by the buyer to the seller (or seller's agent) i) on the expiration date, in the case of a European style option, (ii) on each bermuda option exercise date and the expiration date, in the case of a Bermudan style option and (iii) all days that are exercise business days from and including the commencement date to, and including, the expiration date, in the case of an American style

option.

expirationTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The latest time for expiration on expirationDate.

partialExercise (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_PartialExercise)

- As defined in the 2000 ISDA Definitions, Section 12.3. Partial Exercise, the buyer of the option has the right to exercise all or less than all the notional amount of the underlying swap on the expiration date, but may not exercise less than the minimum notional amount, and if an integral multiple amount is specified, the notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.

exerciseFee (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_ExerciseFee)

- A fee to be paid on exercise. This could be represented as an amount or a rate and notional reference on which to apply the rate.

Used by:

- europeanExercise

DTD Fragment:

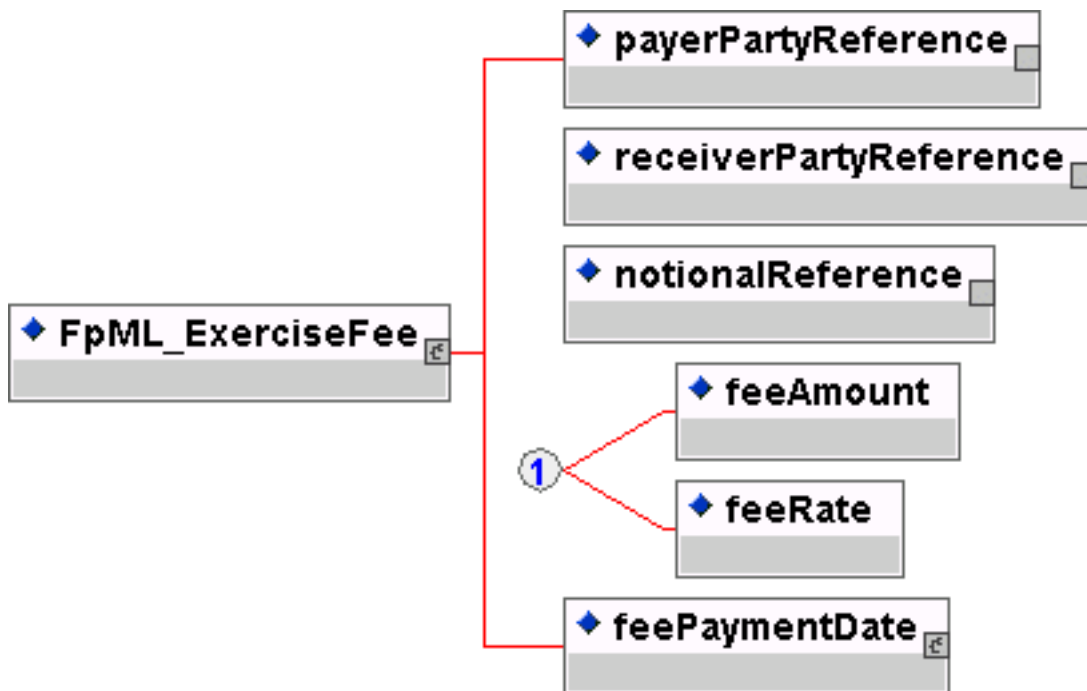
```
<!ENTITY % FpML_EuropeanExercise "expirationDate , relevantUnderlyingDate? , earliestExerciseTime ,
expirationTime , partialExercise? , exerciseFee?">
```

FpML_ExerciseFee

Description:

An entity to define a fee to be payable on exercise of an option. This fee may be defined as an amount or a percentage of the notional exercised.

Figure:



Contents:

payerPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

receiverPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

notionalReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to the associated notional schedule defined elsewhere in the document.

Either

feeAmount (exactly one occurrence; of type *decimal*)

- The amount of fee to be paid on exercise. The currency of this fee is the currency of the referenced notional

Or

feeRate (exactly one occurrence; of type *decimal*)

- A fee represented as a percentage of some referenced notional

feePaymentDate (exactly one occurrence; contains the sub-element(s) defined by exactly one

occurrence of the entity FpML_RelativeDateOffset)

- The date on which exercise fees will be paid. It can be specified as a relative date.

Used by:

- exerciseFee

DTD Fragment:

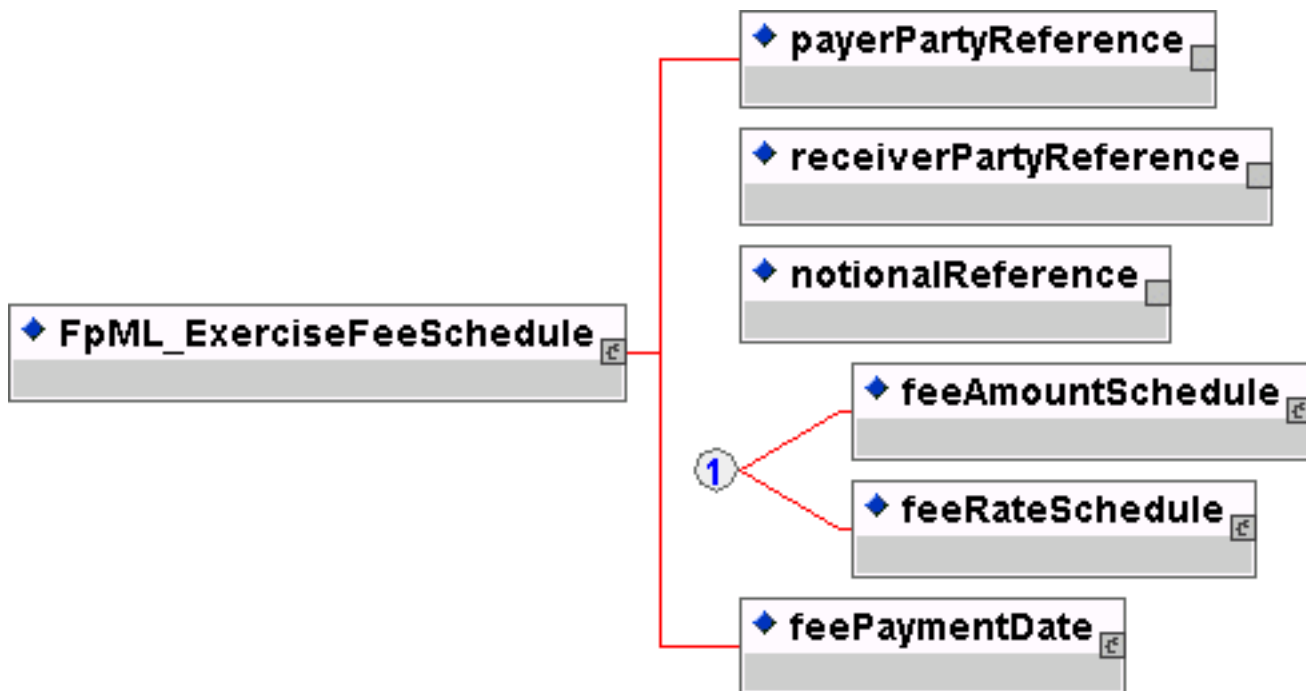
```
<!ENTITY % FpML_ExerciseFee "payerPartyReference , receiverPartyReference , notionalReference ,  
(feeAmount | feeRate) , feePaymentDate">
```


FpML_ExerciseFeeSchedule

Description:

An entity to define a fee or schedule of fees to be payable on exercise of an option. This fee may be defined as an amount or a percentage of the notional exercised.

Figure:



Contents:

payerPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

receiverPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

notionalReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to the associated notional schedule defined elsewhere in the document.

Either

feeAmountSchedule (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AmountSchedule)

- A schedule of fee amounts to be paid on exercise. The currency of this fee is the currency of the referenced notional

Or

feeRateSchedule (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_Schedule)

- A schedule of rates used to calculate an exercise fee based on the referenced notional.

feePaymentDate (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_RelativeDateOffset)

- The date on which exercise fees will be paid. It can be specified as a relative date.

Used by:

- exerciseFeeSchedule

DTD Fragment:

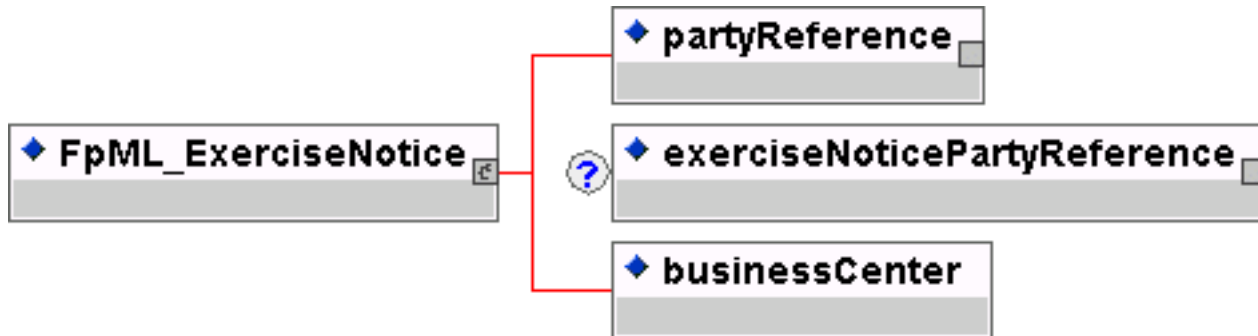
```
<!ENTITY % FpML_ExerciseFeeSchedule "payerPartyReference , receiverPartyReference ,  
notionalReference , (feeAmountSchedule | feeRateSchedule) , feePaymentDate">
```

FpML_ExerciseNotice

Description:

An entity to define to whom and where notice of exercise should be given. The partyReference refers to one of the principal parties of the trade. If present the exerciseNoticePartyReference refers to a party, other than the principal party, to whom notice should be given.

Figure:



Contents:

partyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document. The party referenced has allocated the trade identifier.

exerciseNoticePartyReference (zero or one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document. The party referenced is the party to which notice of exercise should be given by the buyer.

businessCenter (exactly one occurrence; of type *string*, an enumerated domain value defined by *businessCenterScheme*)

- A code identifying a financial business center location. A list of business centers may be ordered in the document alphabetically based on business center code. An FpML document containing an unordered business center list is still regarded as a conformant document.

Used by:

- exerciseNotice

DTD Fragment:

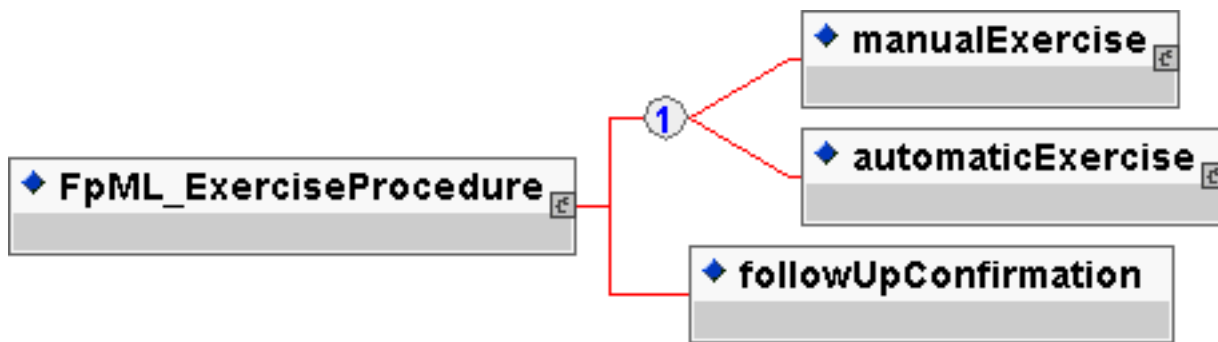
```
<!ENTITY % FpML_ExerciseNotice "partyReference , exerciseNoticePartyReference? , businessCenter">
```

FpML_ExerciseProcedure

Description:

An entity to describe how notice of exercise should be given. This can either be manual or automatic.

Figure:



Contents:

Either

manualExercise (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_ManualExercise)

- Specifies that the notice of exercise must be given by the buyer to the seller or seller's agent.

Or

automaticExercise (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AutomaticExercise)

- If automatic exercise is specified then the notional amount of the underlying swap, not previously exercised under the swaption, will be automatically exercised at the expiration time on the expiration date if at such time the buyer is in-the-money, provided that the difference between the settlement rate and the fixed rate under the relevant underlying swap is not less than the specified thresholdRate. The term In-the-money is assumed to have the meaning defined in the 2000 ISDA Definitions, Section 17.4. In-the-money.

followUpConfirmation (exactly one occurrence; of type *boolean*)

- A flag to indicate whether follow-up confirmation of exercise (written or electronic) is required following telephonic notice by the buyer to the seller or seller's agent.

Used by:

- exerciseProcedure

DTD Fragment:

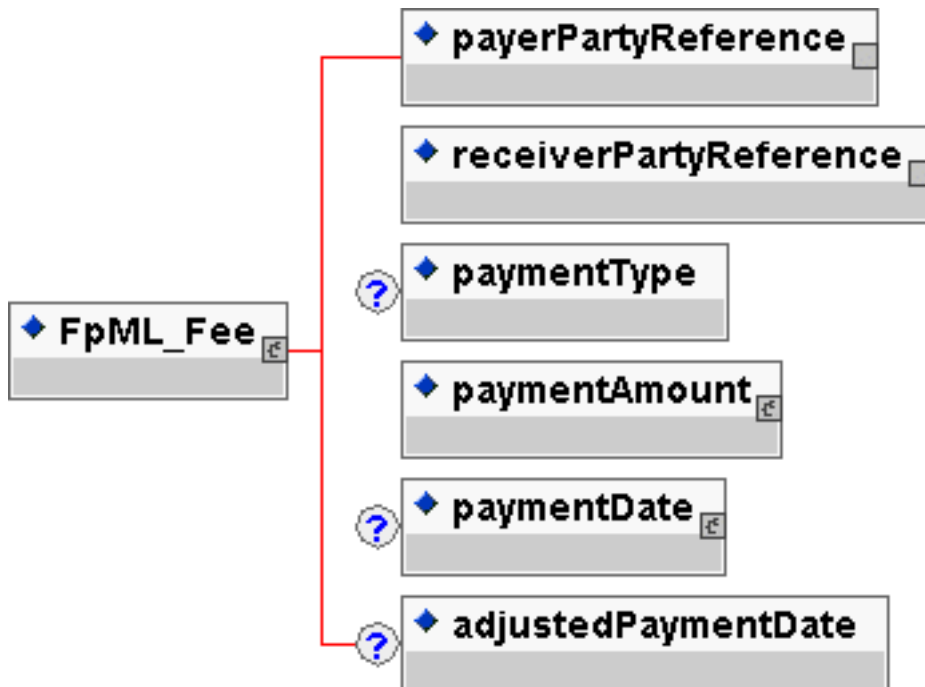
```
<!ENTITY % FpML_ExerciseProcedure "(manualExercise | automaticExercise) , followUpConfirmation">
```

FpML_Fee

Description:

An entity for defining additional payments associated with a trade which are not defined as part of the stream payments. It may be used to define additional payments between the principal parties involved in the trade or other third parties such as a broker. This entity inherits from a base entity, FpML_Payment.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_Payment)

- An entity for defining a payment between two parties.

paymentType (zero or one occurrence; of type *string*, an enumerated domain value defined by *paymentTypeScheme*)

- A classification of the type of fee or additional payment, e.g. brokerage, upfront fee etc. FpML does not define domain values for this element.

Used by:

- additionalPayment
- otherPartyPayment

DTD Fragment:

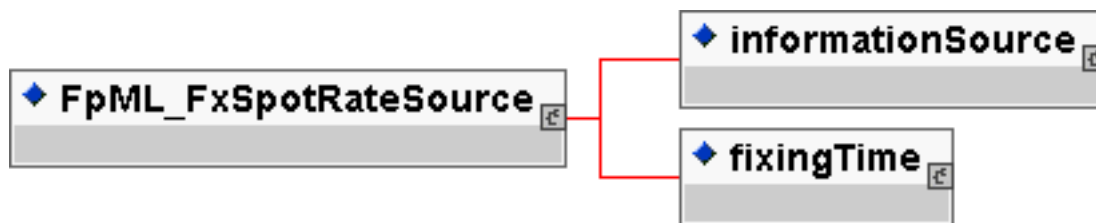
```
<!ENTITY % FpML_Fee "%FpML_Payment; , paymentType?">
```

FpML_FxSpotRateSource

Description:

An entity to define the source and time for an fx rate.

Figure:



Contents:

informationSource (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_InformationSource)

- The information source where a published or displayed market rate will be obtained, e.g. Telerate Page 3750.

fixingTime (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenterTime)

- The time at which the spot currency exchange rate will be observed. It is specified as a time in a specific business center, e.g. 11:00 am London time.

Used by:

- fxSpotRateSource

DTD Fragment:

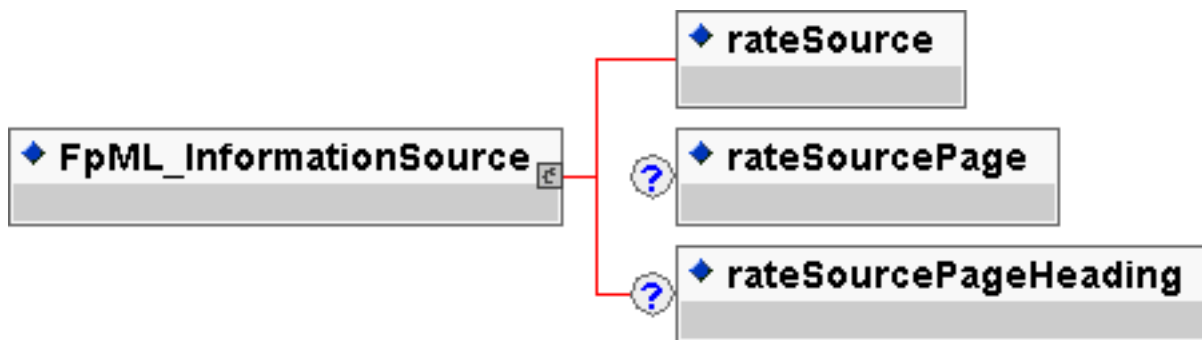
```
<!ENTITY % FpML_FxSpotRateSource "informationSource , fixingTime">
```

FpML_InformationSource

Description:

An entity to define the source for a piece of information (eg a rate refix or a fx fixing).

Figure:



Contents:

rateSource (exactly one occurrence; of type *string*, an enumerated domain value defined by *informationProviderScheme*)

- An information source for obtaining a market rate. For example Bloomberg, Reuters, Telerate etc.

rateSourcePage (zero or one occurrence; of type *string*, an enumerated domain value defined by *rateSourcePageScheme*)

- A specific page for the rate source for obtaining a market rate.

rateSourcePageHeading (zero or one occurrence; of type *string*)

- The specific information source page for obtaining a market rate. For example, 3750 (Telerate), LIBO (Reuters) etc.

Used by:

- informationSource
- primaryRateSource
- secondaryRateSource

DTD Fragment:

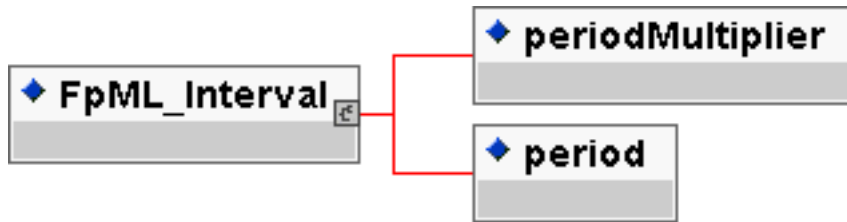
```
<!ENTITY % FpML_InformationSource "rateSource , rateSourcePage? , rateSourcePageHeading?">
```

FpML_Interval

Description:

An entity for defining a time interval or offset, e.g. one day, three months. Used for specifying frequencies at which events occur, the tenor of a floating rate or an offset relative to another date.

Figure:



Contents:

periodMultiplier (exactly one occurrence; of type *integer*)

- A time period multiplier, e.g. 1, 2 or 3 etc. A negative value can be used when specifying an offset relative to another date, e.g. -2 days. If the period value is T (Term) then periodMultiplier must contain the value 1.

period (exactly one occurrence; of type *string*, an enumerated domain value defined by *periodScheme*)

- A time period, e.g. a day, week, month, year or term of the stream. If the periodMultiplier value is 0 (zero) then period must contain the value D (day).

Used by:

- FpML_CalculationPeriodFrequency
- FpML_Offset
- FpML_ResetFrequency
- indexTenor
- paymentFrequency
- quotedTenor
- stepFrequency

DTD Fragment:

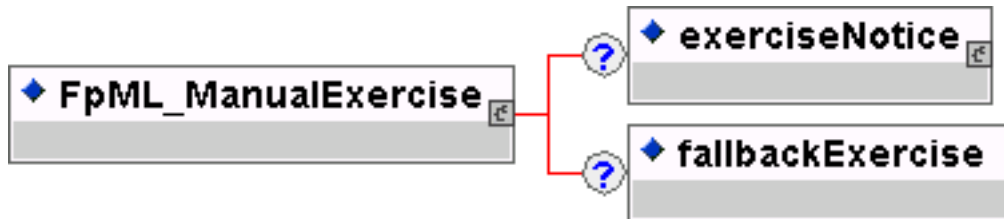
```
<!ENTITY % FpML_Interval "periodMultiplier , period">
```


FpML_ManualExercise

Description:

An entity to define manual exercise. ie that option buyer counterparty must give notice to the option seller of exercise.

Figure:



Contents:

exerciseNotice (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_ExerciseNotice)

- Definition of the party to whom notice of exercise should be given.

fallbackExercise (zero or one occurrence; of type *boolean*)

- If fallback exercise is specified then the notional amount of the underlying swap, not previously exercised under the swaption, will be automatically exercised at the expiration time on the expiration date if at such time the buyer is in-the-money, provided that the difference between the settlement rate and the fixed rate under the relevant underlying swap is not less than one tenth of a percentage point (0.10% or 0.001). The term In-the-money is assumed to have the meaning defined in the 2000 ISDA Definitions, Section 17.4. In-the-money.

Used by:

- manualExercise

DTD Fragment:

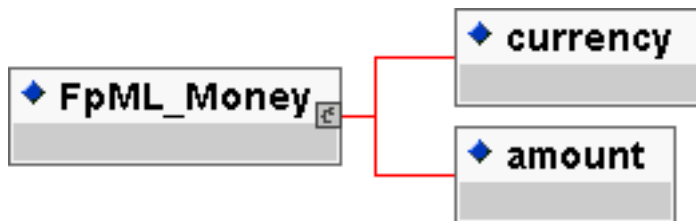
```
<!ENTITY % FpML_ManualExercise "exerciseNotice? , fallbackExercise?">
```

FpML_Money

Description:

An entity for defining a currency amount.

Figure:



Contents:

currency (exactly one occurrence; of type *string*, an enumerated domain value defined by *currencyScheme*)

- The currency in which an amount is denominated.

amount (exactly one occurrence; of type *decimal*)

- The monetary quantity in currency units.

Used by:

- FpML_FXOptionPayout
- callCurrencyAmount
- notional
- paymentAmount
- premiumAmount
- putCurrencyAmount
- splitSettlementAmount
- stubAmount

DTD Fragment:

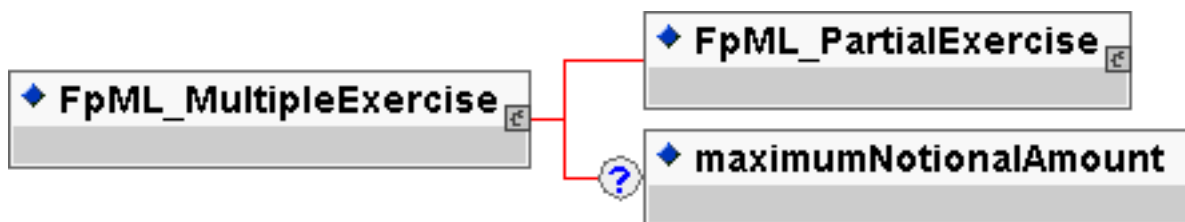
```
<!ENTITY % FpML_Money "currency , amount">
```

FpML_MultipleExercise

Description:

An entity to define multiple exercise. As defined in the 2000 ISDA Definitions, Section 12.4. Multiple Exercise, the buyer of the option has the right to exercise all or less than all the unexercised notional amount of the underlying swap on one or more days in the exercise period, but on any such day may not exercise less than the minimum notional amount or more than the maximum notional amount, and if an integral multiple amount is specified, the notional amount exercised must be equal to, or be an integral multiple of, the integral multiple amount.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_PartialExercise)

- An entity to define partial exercise. As defined in the 2000 ISDA Definitions, Section 12.3 Partial Exercise, the buyer of the option may exercise all or less than all the notional amount of the underlying swap but may not be less than the minimum notional amount (if specified) and must be an integral multiple of the integral multiple amount if specified.

maximumNotionalAmount (zero or one occurrence; of type *decimal*)

- The maximum notional amount that can be exercised on a given exercise date.

Used by:

- multipleExercise

DTD Fragment:

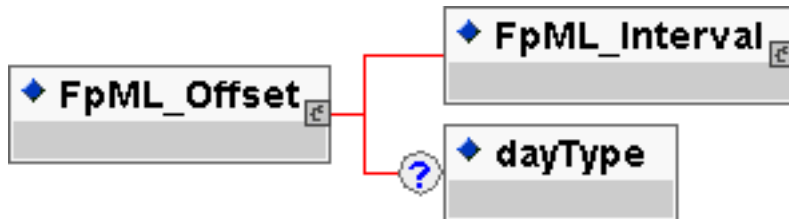
```
<!ENTITY % FpML_MultipleExercise "%FpML_PartialExercise; , maximumNotionalAmount?">
```

FpML_Offset

Description:

An entity for defining an offset used in calculating a new date relative to a reference date. Currently, the only offsets defined are expected to be expressed as either calendar or business day offsets. This entity inherits from a base entity, FpML_Interval.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_Interval)

- An entity for defining a time interval or offset, e.g. one day, three months. Used for specifying frequencies at which events occur, the tenor of a floating rate or an offset relative to another date.

dayType (zero or one occurrence; of type *string*, an enumerated domain value defined by *dayTypeScheme*)

- In the case of an offset specified as a number of days, this element defines whether consideration is given as to whether a day is a good business day or not. If a day type of business days is specified then non-business days are ignored when calculating the offset. The financial business centers to use for determination of business days are implied by the context in which this element is used. This element must only be included when the offset is specified as a number of days. If the offset is zero days then the dayType element should not be included.

Used by:

- FpML_RelativeDateOffset
- paymentDaysOffset
- rateCutOffDaysOffset

DTD Fragment:

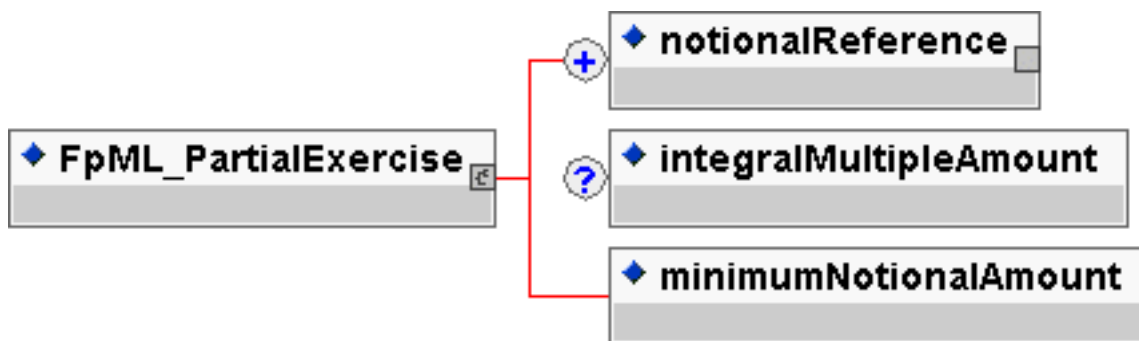
```
<!ENTITY % FpML_Offset "(%FpML_Interval; , dayType?)">
```

FpML_PartialExercise

Description:

An entity to define partial exercise. As defined in the 2000 ISDA Definitions, Section 12.3 Partial Exercise, the buyer of the option may exercise all or less than all the notional amount of the underlying swap but may not be less than the minimum notional amount (if specified) and must be an integral multiple of the integral multiple amount if specified.

Figure:



Contents:

notionalReference (one or more occurrences; an *empty* element containing an *href* attribute)

- A pointer style reference to the associated notional schedule defined elsewhere in the document.

integralMultipleAmount (zero or one occurrence; of type *decimal*)

- A notional amount which restricts the amount of notional that can be exercised when partial exercise or multiple exercise is applicable. The integral multiple amount defines a lower limit of notional that can be exercised and also defines a unit multiple of notional that can be exercised, i.e. only integer multiples of this amount can be exercised.

minimumNotionalAmount (exactly one occurrence; of type *decimal*)

- The minimum notional amount that can be exercised on a given exercise date. See `multipleExercise`.

Used by:

- `FpML_MultipleExercise`
- `partialExercise`

DTD Fragment:

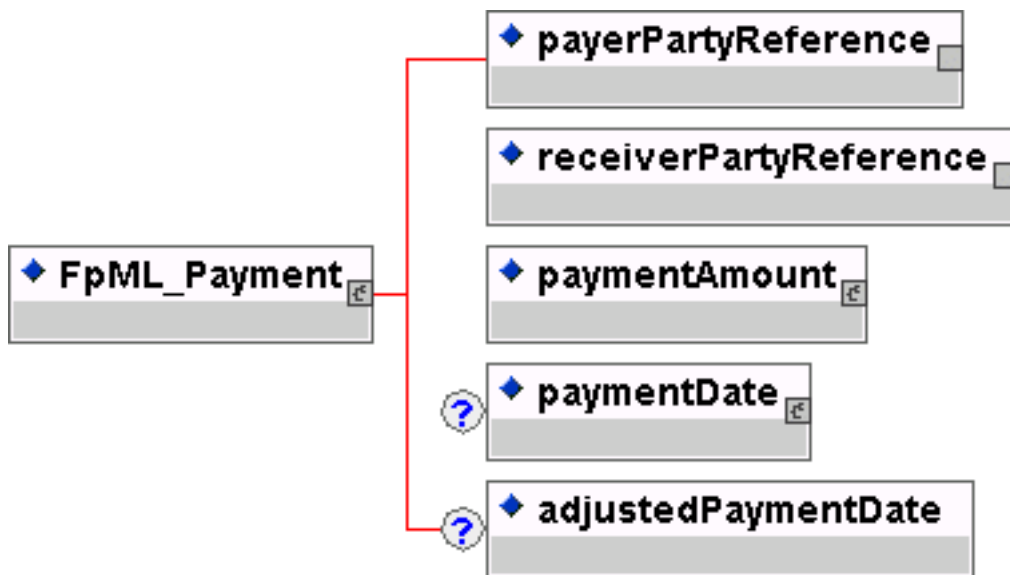
```
<!ENTITY % FpML_PartialExercise "notionalReference+ , integralMultipleAmount? ,
minimumNotionalAmount">
```

FpML_Payment

Description:

An entity for defining a payment between two parties.

Figure:



Contents:

payerPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

receiverPartyReference (exactly one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a party identifier defined elsewhere in the document.

paymentAmount (exactly one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_Money)

- The currency amount of the payment.

paymentDate (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_AdjustableDate)

- The payment date. This date is subject to adjustment in accordance with any applicable business day convention.

adjustedPaymentDate (zero or one occurrence; of type *date*)

- The adjusted payment date. This date should already be adjusted for any applicable business day convention.

Used by:

- FpML_CurrencyFlow
- FpML_Fee
- payment
- premium

DTD Fragment:

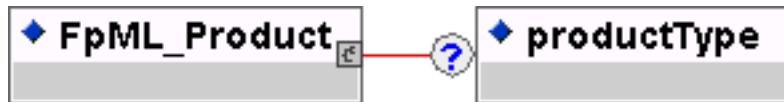
```
<!ENTITY % FpML_Payment "payerPartyReference , receiverPartyReference , paymentAmount ,  
paymentDate? , adjustedPaymentDate?">
```

FpML_Product

Description:

The base entity which all FpML products extend.

Figure:



Contents:

productType (zero or one occurrence; of type *string*, an enumerated domain value defined by *productTypeScheme*)

- A classification of the type of product. Fpml does not define a domain of values for this element.

Used by:

- FpML_BulletPayment
- FpML_CapFloor
- FpML_Fra
- FpML_FXAverageRateOption
- FpML_FXDigitalOption
- FpML_FXLeg
- FpML_FXOptionLeg
- FpML_FXSwap
- FpML_Strategy
- FpML_Swap
- FpML_Swaption

DTD Fragment:

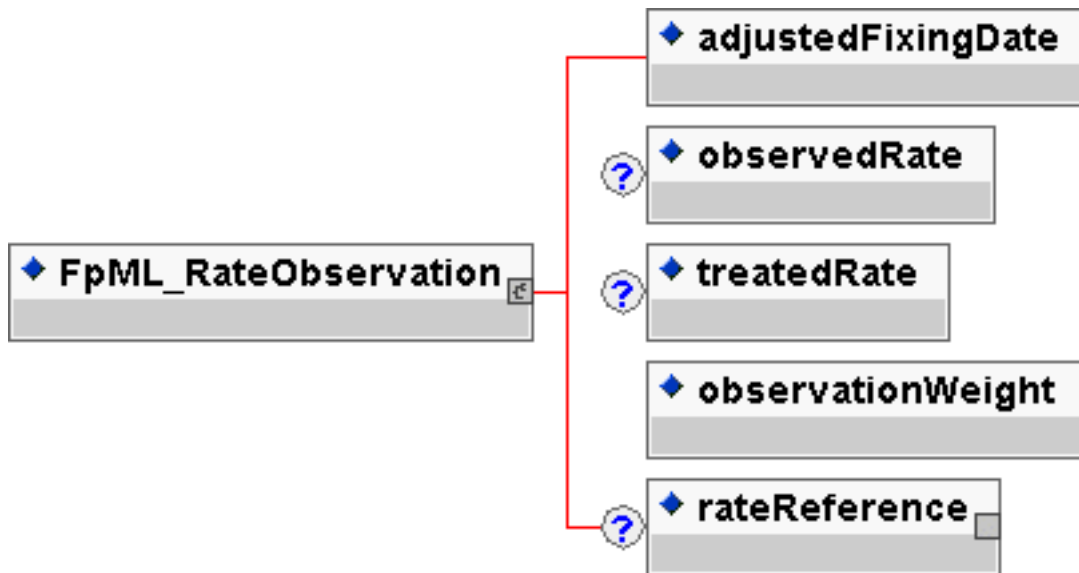
```
<!ENTITY % FpML_Product "productType?">
```


FpML_RateObservation

Description:

An entity for defining parameters associated with an individual rate observation or fixing. This entity forms part of the cashflows representation of a stream.

Figure:



Contents:

adjustedFixingDate (exactly one occurrence; of type *date*)

- The adjusted fixing date, i.e. the actual date the rate is observed. This date should already be adjusted for any applicable business day convention.

observedRate (zero or one occurrence; of type *decimal*)

- The actual observed rate before any required rate treatment is applied, e.g. before converting a rate quoted on a discount basis to an equivalent yield. An observed rate of 5% would be represented as 0.05.

treatedRate (zero or one occurrence; of type *decimal*)

- The observed rate after any required rate treatment is applied. A treated rate of 5% would be represented as 0.05.

observationWeight (exactly one occurrence; of type *positiveInteger*)

- The number of days weighting to be associated with the rate observation, i.e. the number of days such rate is in effect. This is applicable in the case of a weighted average method of calculation where more than one reset date is established for a single calculation period.

rateReference (zero or one occurrence; an *empty* element containing an *href* attribute)

- A pointer style reference to a floating rate component defined as part of a stub calculation period amount component. It is only required when it is necessary to distinguish two rate observations for the same fixing date which could occur when linear interpolation of two different rates occurs for a stub calculation period.

Used by:

- rateObservation

DTD Fragment:

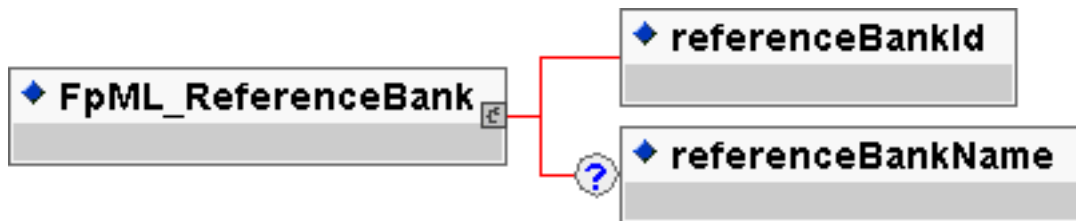
```
<!ENTITY % FpML_RateObservation "adjustedFixingDate , observedRate? , treatedRate? ,  
observationWeight , rateReference?">
```

FpML_ReferenceBank

Description:

An entity to describe institution (party) identified by means of a coding scheme and an optional name.

Figure:



Contents:

referenceBankId (exactly one occurrence; of type *string*, an enumerated domain value defined by *referenceBankIdScheme*)

- An institution (party) identifier, e.g. a bank identifier code (BIC).

referenceBankName (zero or one occurrence; of type *string*)

- The name of the institution (party). A free format string. FpML does not define usage rules for the element.

Used by:

- referenceBank

DTD Fragment:

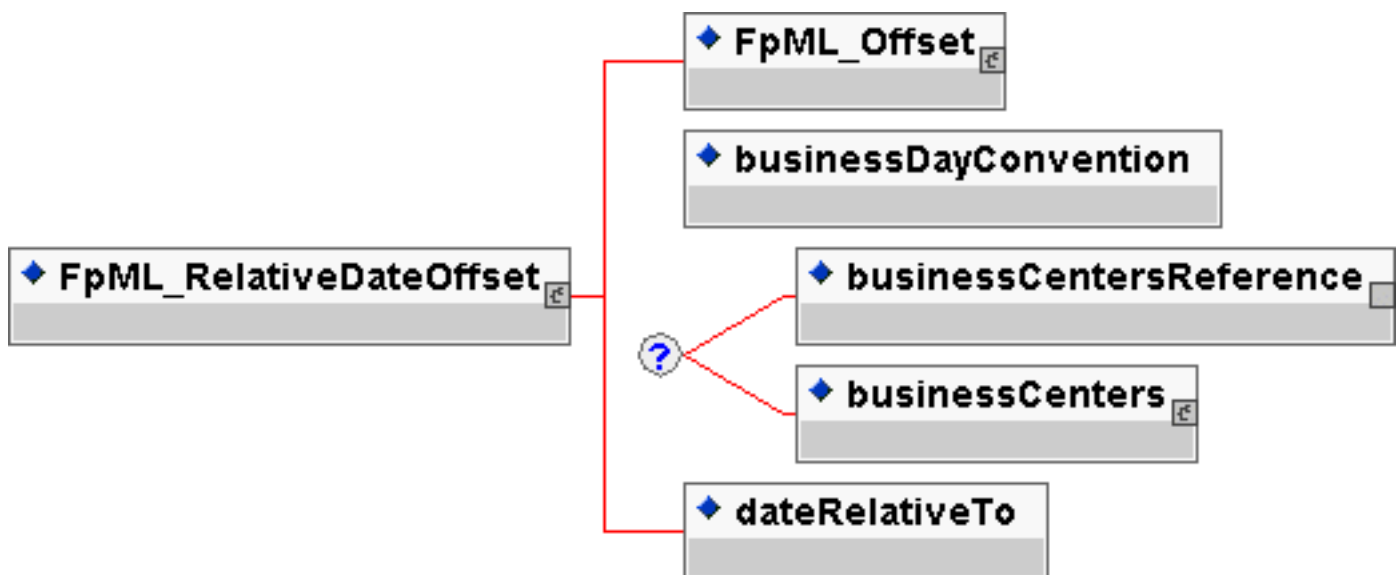
```
<!ENTITY % FpML_ReferenceBank "referenceBankId , referenceBankName?">
```

FpML_RelativeDateOffset

Description:

An entity for defining a date (referred to as the derived date) as a relative offset from another date (referred to as the anchor date). If the anchor date is itself an adjustable date then the offset is assumed to be calculated from the adjusted anchor date. A number of different scenarios can be supported, namely; 1) the derived date may simply be a number of calendar periods (days, weeks, months or years) preceding or following the anchor date; 2) the unadjusted derived date may be a number of calendar periods (days, weeks, months, years) preceding or following the anchor date with the resulting unadjusted derived date subject to adjustment in accordance with a specified business day convention, i.e. the derived date must fall on a good business day; 3) the derived date may be a number of business days preceding or following the anchor date. Note that the `businessDayConvention` element specifies any required adjustment to the unadjusted derived date. A negative or positive value in the `periodMultiplier` element indicates whether the unadjusted derived date precedes or follows the anchor date. The `businessDayConvention` element should contain a value of `NONE` if the `dayType` element contains a value of `Business` (since specifying a negative or positive business days offset would already guarantee that the derived date would fall on a good business day in the specified business centers).

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity `FpML_Offset`)

- An entity for defining an offset used in calculating a new date relative to a reference date. Currently, the only offsets defined are expected to be expressed as either calendar or business day offsets. This entity inherits from a base entity, `FpML_Interval`.

businessDayConvention (exactly one occurrence; of type *string*, an enumerated domain value defined by *businessDayConventionScheme*)

- The convention for adjusting a date if it would otherwise fall on a day that is not a business day.

Zero or one occurrence of either

businessCentersReference (exactly one occurrence; an *empty* element containing an *href*

attribute)

- A pointer style reference to a set of financial business centers defined elsewhere in the document. This set of business centers is used to determine whether a particular day is a business day or not.

Or

businessCenters (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_BusinessCenters)

- A container for a set of financial business centers. This set of business centers is used to determine whether a day is a business day or not.

dateRelativeTo (exactly one occurrence; of type *string*, an enumerated domain value defined by *dateRelativeToScheme*)

- Specifies the anchor date. This element also carries an href attribute. The href attribute value will be a pointer style reference to the element or component elsewhere in the document where the anchor date is defined.

Used by:

- FpML_RelativeDates
- cashSettlementValuationDate
- feePaymentDate
- fixingDateOffset
- fixingDates
- relativeDate
- varyingNotionalFixingDates
- varyingNotionalInterimExchangePaymentDates

DTD Fragment:

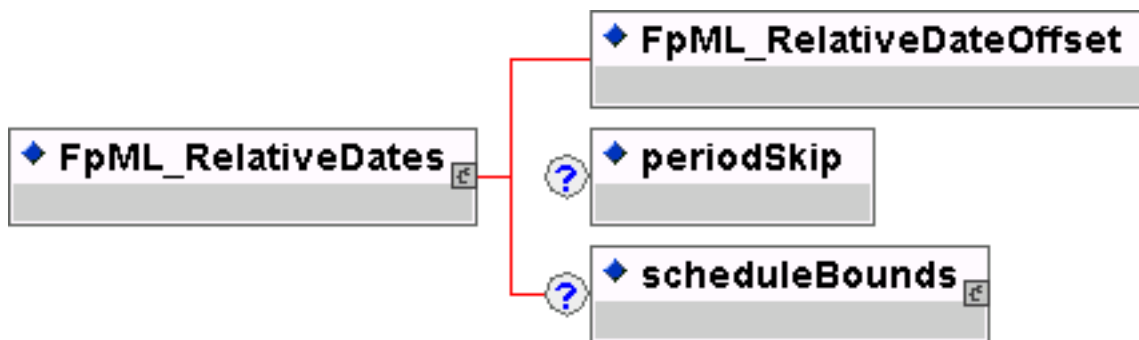
```
<!ENTITY % FpML_RelativeDateOffset "(%FpML_Offset; , businessDayConvention ,
(businessCentersReference | businessCenters)? , dateRelativeTo)">
```

FpML_RelativeDates

Description:

An entity to define a set of dates defined as relative to another set of dates.

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_RelativeDateOffset)

- An entity for defining a date (referred to as the derived date) as a relative offset from another date (referred to as the anchor date). If the anchor date is itself an adjustable date then the offset is assumed to be calculated from the adjusted anchor date. A number of different scenarios can be supported, namely; 1) the derived date may simply be a number of calendar periods (days, weeks, months or years) preceding or following the anchor date; 2) the unadjusted derived date may be a number of calendar periods (days, weeks, months, years) preceding or following the anchor date with the resulting unadjusted derived date subject to adjustment in accordance with a specified business day convention, i.e. the derived date must fall on a good business day; 3) the derived date may be a number of business days preceding or following the anchor date. Note that the `businessDayConvention` element specifies any required adjustment to the unadjusted derived date. A negative or positive value in the `periodMultiplier` element indicates whether the unadjusted derived date precedes or follows the anchor date. The `businessDayConvention` element should contain a value of `NONE` if the `dayType` element contains a value of `Business` (since specifying a negative or positive business days offset would already guarantee that the derived date would fall on a good business day in the specified business centers).

periodSkip (zero or one occurrence; of type *positiveInteger*)

- The number of periods in the referenced date schedule that are between each date in the relative date schedule. Thus a skip of 2 would mean that dates are relative to every second date in the referenced schedule. If present this should have a value greater than 1.

scheduleBounds (zero or one occurrence; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_DateRange)

- The first and last dates of a schedule. This can be used to restrict the range of values in a reference series of dates.

Used by:

- relativeDates

DTD Fragment:

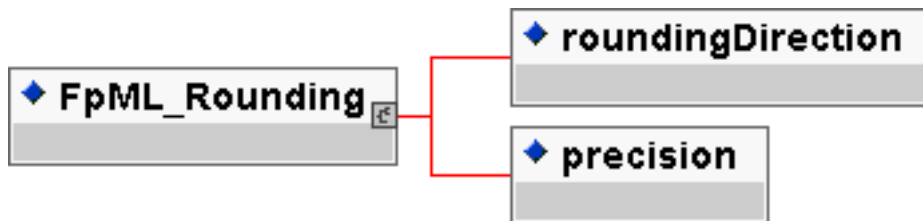
```
<!ENTITY % FpML_RelativeDates "(%FpML_RelativeDateOffset; , periodSkip? , scheduleBounds?)">
```

FpML_Rounding

Description:

An entity for defining a rounding direction and precision to be used in the rounding of a rate.

Figure:



Contents:

roundingDirection (exactly one occurrence; of type *string*, an enumerated domain value defined by *roundingDirectionScheme*)

- Specifies the rounding direction.

precision (exactly one occurrence; of type *nonNegativeInteger*)

- Specifies the rounding precision in terms of a number of decimal places. Note how a percentage rate rounding of 5 decimal places is expressed as a rounding precision of 7 in the FpML document since the percentage is expressed as a decimal, e.g. 9.876543% (or 0.09876543) being rounded to the nearest 5 decimal places is 9.87654% (or 0.0987654).

Used by:

- finalRateRounding

DTD Fragment:

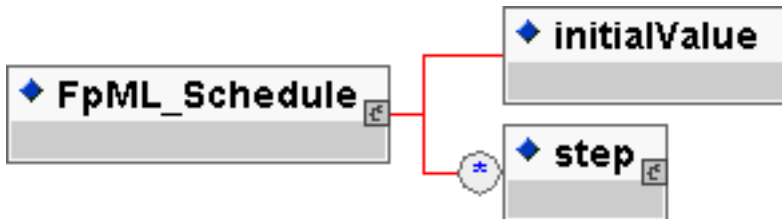
```
<!ENTITY % FpML_Rounding "roundingDirection , precision">
```


FpML_Schedule

Description:

An entity for defining a schedule of rate or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

Figure:



Contents:

initialValue (exactly one occurrence; of type *decimal*)

- The initial rate or amount, as the case may be. An initial rate of 5% would be represented as 0.05.

step (zero or more occurrences; contains the sub-element(s) defined by exactly one occurrence of the entity FpML_Step)

- The schedule of step date and value pairs. On each step date the associated step value becomes effective. A list of steps may be ordered in the document by ascending step date. An FpML document containing an unordered list of steps is still regarded as a conformant document.

Used by:

- FpML_AmountSchedule
- FpML_StrikeSchedule
- feeAmountSchedule
- feeRateSchedule
- fixedRateSchedule
- floatingRateMultiplierSchedule
- spreadSchedule

DTD Fragment:

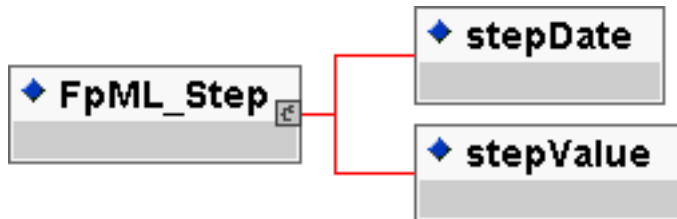
```
<!ENTITY % FpML_Schedule "initialValue , step*">
```

FpML_Step

Description:

An entity for defining a step date and step value pair. These step definitions are used to define varying rate or amount schedules, e.g. a notional amortization or a step-up coupon schedule.

Figure:



Contents:

stepDate (exactly one occurrence; of type *date*)

- The date on which the associated stepValue becomes effective. This day may be subject to adjustment in accordance with a business day convention.

stepValue (exactly one occurrence; of type *decimal*)

- The rate or amount which becomes effective on the associated stepDate. A rate of 5% would be represented as 0.05.

Used by:

- step

DTD Fragment:

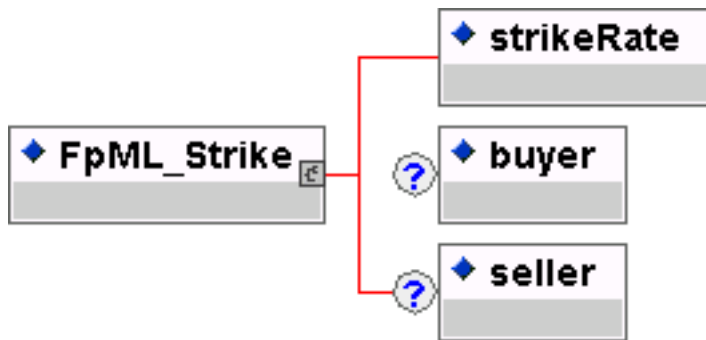
```
<!ENTITY % FpML_Step "stepDate , stepValue">
```

FpML_Strike

Description:

TBA

Figure:



Contents:

strikeRate (exactly one occurrence; of type *decimal*)

- The rate for a cap or floor.

buyer (zero or one occurrence; of type *string*, an enumerated domain value defined by *payerReceiverScheme*)

- The buyer of the option

seller (zero or one occurrence; of type *string*, an enumerated domain value defined by *payerReceiverScheme*)

- The party that has sold.

Used by:

- capRate
- floorRate

DTD Fragment:

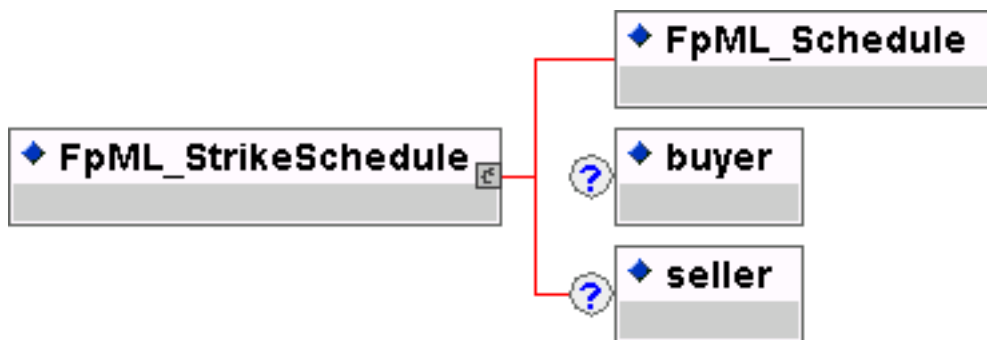
```
<!ENTITY % FpML_Strike "strikeRate , buyer? , seller?">
```

FpML_StrikeSchedule

Description:

TBA

Figure:



Contents:

inherited element(s) (this entity inherits the element(s) defined by exactly one occurrence of the entity FpML_Schedule)

- An entity for defining a schedule of rate or amounts in terms of an initial value and then a series of step date and value pairs. On each step date the rate or amount changes to the new step value. The series of step date and value pairs are optional. If not specified, this implies that the initial value remains unchanged over time.

buyer (zero or one occurrence; of type *string*, an enumerated domain value defined by *payerReceiverScheme*)

- The buyer of the option

seller (zero or one occurrence; of type *string*, an enumerated domain value defined by *payerReceiverScheme*)

- The party that has sold.

Used by:

- capRateSchedule
- floorRateSchedule

DTD Fragment:

```
<!ENTITY % FpML_StrikeSchedule "(%FpML_Schedule; , buyer? , seller?)">
```