



Financial products Markup Language

FpML - Repo Component Definitions

Version: 4.3

This Version:

<http://www.fpml.org/spec/2007/wd-fpml-4-3-2007-05-14>

Latest Version:

<http://www.fpml.org/spec/2007/wd-fpml-4-3-2007-05-14>

Previous Version:

<http://www.fpml.org/spec/2007/wd-fpml-4-3-2007-02-05/>

Errata For This Version:

<http://www.fpml.org/spec/errata/wd-fpml-4-3-2007-05-14-errata.html>

Document built

Copyright (c) 1999 - 2007 by International Swaps and Derivatives Association, Inc.

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

FpML is a registered trademark of the International Swaps and Derivatives Association, Inc.

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

The FpML specifications provided are without warranty of any kind, either expressed or implied, including, without limitation, warranties that FpML, or the FpML specifications are free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the specifications is with you. Should any of the FpML specifications prove defective in any respect, you assume the cost of any necessary servicing or repair. Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall ISDA, any of its members, or any distributor of documents or software containing any of the FpML specifications, or any supplier of any of such parties, be liable to you or any other person for any indirect, special, incidental, or consequential damages of any character including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages.

Table Of Contents

1	Global Simple Types	12
1.1	DeliveryMethodEnum	13
1.1.1	Description:	13
1.1.2	Contents:	13
1.1.3	Used by:	13
1.1.4	Derived Types:	13
1.1.5	Schema Fragment:	13
1.2	MarginTypeEnum	14
1.2.1	Description:	14
1.2.2	Contents:	14
1.2.3	Used by:	14
1.2.4	Derived Types:	14
1.2.5	Schema Fragment:	14
1.3	RepoDurationEnum	15
1.3.1	Description:	15
1.3.2	Contents:	15
1.3.3	Used by:	15
1.3.4	Derived Types:	15
1.3.5	Schema Fragment:	15
2	Global Complex Types	17
2.1	AdjustableOffset	18
2.1.1	Description:	18
2.1.2	Contents:	18
2.1.3	Used by:	18
2.1.4	Derived Types:	18
2.1.5	Figure:	18
2.1.6	Schema Fragment:	18
2.2	AtomicSettlementTransfer	19
2.2.1	Description:	19
2.2.2	Contents:	19
2.2.3	Used by:	19
2.2.4	Derived Types:	19
2.2.5	Figure:	19
2.2.6	Schema Fragment:	19
2.3	Attribution	20
2.3.1	Description:	20
2.3.2	Contents:	20
2.3.3	Used by:	20
2.3.4	Derived Types:	20
2.3.5	Figure:	20
2.3.6	Schema Fragment:	20
2.4	Attributions	22
2.4.1	Description:	22
2.4.2	Contents:	22
2.4.3	Used by:	22
2.4.4	Derived Types:	22
2.4.5	Figure:	22
2.4.6	Schema Fragment:	22
2.5	AttributionType	24
2.5.1	Description:	24
2.5.2	Contents:	24
2.5.3	Used by:	24
2.5.4	Derived Types:	24
2.5.5	Figure:	24
2.5.6	Schema Fragment:	24
2.6	CashRepricingEvent	25
2.6.1	Description:	25
2.6.2	Contents:	25
2.6.3	Used by:	25
2.6.4	Derived Types:	25

2.6.5	Figure:	25
2.6.6	Schema Fragment:	25
2.7	CashTransfer	26
2.7.1	Description:	26
2.7.2	Contents:	26
2.7.3	Used by:	26
2.7.4	Derived Types:	26
2.7.5	Figure:	26
2.7.6	Schema Fragment:	26
2.8	CollateralSubstitutionEvent	28
2.8.1	Description:	28
2.8.2	Contents:	28
2.8.3	Used by:	28
2.8.4	Derived Types:	28
2.8.5	Figure:	28
2.8.6	Schema Fragment:	28
2.9	CollateralValuation	30
2.9.1	Description:	30
2.9.2	Contents:	30
2.9.3	Used by:	30
2.9.4	Derived Types:	30
2.9.5	Figure:	30
2.9.6	Schema Fragment:	30
2.10	CouponEvent	31
2.10.1	Description:	31
2.10.2	Contents:	31
2.10.3	Used by:	31
2.10.4	Derived Types:	31
2.10.5	Figure:	31
2.10.6	Schema Fragment:	31
2.11	EventReference	33
2.11.1	Description:	33
2.11.2	Contents:	33
2.11.3	Used by:	33
2.11.4	Derived Types:	33
2.11.5	Figure:	33
2.11.6	Schema Fragment:	33
2.12	ForwardRepoTransactionLeg	34
2.12.1	Description:	34
2.12.2	Contents:	34
2.12.3	Used by:	34
2.12.4	Derived Types:	34
2.12.5	Figure:	34
2.12.6	Schema Fragment:	34
2.13	InterestPayoutEvent	35
2.13.1	Description:	35
2.13.2	Contents:	35
2.13.3	Used by:	35
2.13.4	Derived Types:	35
2.13.5	Figure:	35
2.13.6	Schema Fragment:	35
2.14	Margin	36
2.14.1	Description:	36
2.14.2	Contents:	36
2.14.3	Used by:	36
2.14.4	Derived Types:	36
2.14.5	Figure:	36
2.14.6	Schema Fragment:	36
2.15	MarkToMarketEvent	37
2.15.1	Description:	37
2.15.2	Contents:	37
2.15.3	Used by:	37
2.15.4	Derived Types:	37
2.15.5	Figure:	37

2.15.6	Schema Fragment:	37
2.16	MidLifeEvent	39
2.16.1	Description:	39
2.16.2	Contents:	39
2.16.3	Used by:	39
2.16.4	Derived Types:	39
2.16.5	Figure:	39
2.16.6	Schema Fragment:	39
2.17	NetTradeIdentifier	41
2.17.1	Description:	41
2.17.2	Contents:	41
2.17.3	Used by:	41
2.17.4	Derived Types:	41
2.17.5	Figure:	41
2.17.6	Schema Fragment:	41
2.18	PartySettlementTransferInformation	42
2.18.1	Description:	42
2.18.2	Contents:	42
2.18.3	Used by:	42
2.18.4	Derived Types:	42
2.18.5	Figure:	42
2.18.6	Schema Fragment:	42
2.19	RateChangeEvent	43
2.19.1	Description:	43
2.19.2	Contents:	43
2.19.3	Used by:	43
2.19.4	Derived Types:	43
2.19.5	Figure:	43
2.19.6	Schema Fragment:	43
2.20	RateObservationEvent	44
2.20.1	Description:	44
2.20.2	Contents:	44
2.20.3	Used by:	44
2.20.4	Derived Types:	44
2.20.5	Figure:	44
2.20.6	Schema Fragment:	44
2.21	RelativePrice	45
2.21.1	Description:	45
2.21.2	Contents:	45
2.21.3	Used by:	45
2.21.4	Derived Types:	45
2.21.5	Figure:	45
2.21.6	Schema Fragment:	45
2.22	Repo	46
2.22.1	Description:	46
2.22.2	Contents:	46
2.22.3	Used by:	46
2.22.4	Derived Types:	46
2.22.5	Figure:	46
2.22.6	Schema Fragment:	46
2.23	RepoLegId	49
2.23.1	Description:	49
2.23.2	Contents:	49
2.23.3	Used by:	49
2.23.4	Derived Types:	49
2.23.5	Figure:	49
2.23.6	Schema Fragment:	49
2.24	RepoTransactionLeg	50
2.24.1	Description:	50
2.24.2	Contents:	50
2.24.3	Used by:	50
2.24.4	Derived Types:	50
2.24.5	Figure:	50
2.24.6	Schema Fragment:	50

2.25	RepoTransactionLegReference	52
2.25.1	Description:	52
2.25.2	Contents:	52
2.25.3	Used by:	52
2.25.4	Derived Types:	52
2.25.5	Figure:	52
2.25.6	Schema Fragment:	52
2.26	SecurityTransfer	53
2.26.1	Description:	53
2.26.2	Contents:	53
2.26.3	Used by:	53
2.26.4	Derived Types:	53
2.26.5	Figure:	53
2.26.6	Schema Fragment:	53
2.27	SettlementInstructionReference	55
2.27.1	Description:	55
2.27.2	Contents:	55
2.27.3	Used by:	55
2.27.4	Derived Types:	55
2.27.5	Figure:	55
2.27.6	Schema Fragment:	55
2.28	SettlementTransfer	56
2.28.1	Description:	56
2.28.2	Contents:	56
2.28.3	Used by:	56
2.28.4	Derived Types:	56
2.28.5	Figure:	56
2.28.6	Schema Fragment:	56
2.29	SettlementTransferId	58
2.29.1	Description:	58
2.29.2	Contents:	58
2.29.3	Used by:	58
2.29.4	Derived Types:	58
2.29.5	Figure:	58
2.29.6	Schema Fragment:	58
2.30	SettlementTransferIdentifier	59
2.30.1	Description:	59
2.30.2	Contents:	59
2.30.3	Used by:	59
2.30.4	Derived Types:	59
2.30.5	Figure:	59
2.30.6	Schema Fragment:	59
2.31	SettlementTransferProcessingInformation	60
2.31.1	Description:	60
2.31.2	Contents:	60
2.31.3	Used by:	60
2.31.4	Derived Types:	60
2.31.5	Figure:	60
2.31.6	Schema Fragment:	60
2.32	SettlementTransferType	61
2.32.1	Description:	61
2.32.2	Contents:	61
2.32.3	Used by:	61
2.32.4	Derived Types:	61
2.32.5	Figure:	61
2.32.6	Schema Fragment:	61
2.33	StreamId	62
2.33.1	Description:	62
2.33.2	Contents:	62
2.33.3	Used by:	62
2.33.4	Derived Types:	62
2.33.5	Figure:	62
2.33.6	Schema Fragment:	62
2.34	StreamReference	

2.34.1	Description:	63
2.34.2	Contents:	63
2.34.3	Used by:	63
2.34.4	Derived Types:	63
2.34.5	Figure:	63
2.34.6	Schema Fragment:	63
2.35	TradeAndComponentIdentifier	64
2.35.1	Description:	64
2.35.2	Contents:	64
2.35.3	Used by:	64
2.35.4	Derived Types:	64
2.35.5	Figure:	64
2.35.6	Schema Fragment:	64
2.36	TradeComponentIdentifier	65
2.36.1	Description:	65
2.36.2	Contents:	65
2.36.3	Used by:	65
2.36.4	Derived Types:	65
2.36.5	Figure:	65
2.36.6	Schema Fragment:	65
2.37	TradeIdentifierList	66
2.37.1	Description:	66
2.37.2	Contents:	66
2.37.3	Used by:	66
2.37.4	Derived Types:	66
2.37.5	Figure:	66
2.37.6	Schema Fragment:	66
2.38	Transfer	67
2.38.1	Description:	67
2.38.2	Contents:	67
2.38.3	Used by:	67
2.38.4	Derived Types:	67
2.38.5	Figure:	67
2.38.6	Schema Fragment:	67
2.39	TransferId	69
2.39.1	Description:	69
2.39.2	Contents:	69
2.39.3	Used by:	69
2.39.4	Derived Types:	69
2.39.5	Figure:	69
2.39.6	Schema Fragment:	69
2.40	VersionedEventId	70
2.40.1	Description:	70
2.40.2	Contents:	70
2.40.3	Used by:	70
2.40.4	Derived Types:	70
2.40.5	Figure:	70
2.40.6	Schema Fragment:	70
2.41	VersionedRepoLegId	71
2.41.1	Description:	71
2.41.2	Contents:	71
2.41.3	Used by:	71
2.41.4	Derived Types:	71
2.41.5	Figure:	71
2.41.6	Schema Fragment:	71
2.42	VersionedSettlementTransferId	72
2.42.1	Description:	72
2.42.2	Contents:	72
2.42.3	Used by:	72
2.42.4	Derived Types:	72
2.42.5	Figure:	72
2.42.6	Schema Fragment:	72
2.43	VersionedSettlementTransferType	73
2.43.1	Description:	73

2.43.2	Contents:	73
2.43.3	Used by:	73
2.43.4	Derived Types:	73
2.43.5	Figure:	73
2.43.6	Schema Fragment:	73
2.44	VersionedStreamId	74
2.44.1	Description:	74
2.44.2	Contents:	74
2.44.3	Used by:	74
2.44.4	Derived Types:	74
2.44.5	Figure:	74
2.44.6	Schema Fragment:	74
2.45	VersionedTransferId	75
2.45.1	Description:	75
2.45.2	Contents:	75
2.45.3	Used by:	75
2.45.4	Derived Types:	75
2.45.5	Figure:	75
2.45.6	Schema Fragment:	75
3	Global Elements	76
3.1	cashRepricing	77
3.1.1	Description:	77
3.1.2	Contents:	77
3.1.3	Used by:	77
3.1.4	Substituted by:	77
3.1.5	Figure:	77
3.1.6	Schema Fragment:	77
3.2	collateralSubstitution	78
3.2.1	Description:	78
3.2.2	Contents:	78
3.2.3	Used by:	78
3.2.4	Substituted by:	78
3.2.5	Figure:	78
3.2.6	Schema Fragment:	78
3.3	couponEvent	79
3.3.1	Description:	79
3.3.2	Contents:	79
3.3.3	Used by:	79
3.3.4	Substituted by:	79
3.3.5	Figure:	79
3.3.6	Schema Fragment:	79
3.4	interestPayout	80
3.4.1	Description:	80
3.4.2	Contents:	80
3.4.3	Used by:	80
3.4.4	Substituted by:	80
3.4.5	Figure:	80
3.4.6	Schema Fragment:	80
3.5	markToMarketEvent	81
3.5.1	Description:	81
3.5.2	Contents:	81
3.5.3	Used by:	81
3.5.4	Substituted by:	81
3.5.5	Figure:	81
3.5.6	Schema Fragment:	81
3.6	midLifeEvent	82
3.6.1	Description:	82
3.6.2	Contents:	82
3.6.3	Used by:	82
3.6.4	Substituted by:	82
3.6.5	Figure:	82
3.6.6	Schema Fragment:	82
3.7	rateChange	83
3.7.1	Description:	83

3.7.2	Contents:	83
3.7.3	Used by:	83
3.7.4	Substituted by:	83
3.7.5	Figure:	83
3.7.6	Schema Fragment:	83
3.8	rateObservation	84
3.8.1	Description:	84
3.8.2	Contents:	84
3.8.3	Used by:	84
3.8.4	Substituted by:	84
3.8.5	Figure:	84
3.8.6	Schema Fragment:	84
3.9	repo	85
3.9.1	Description:	85
3.9.2	Contents:	85
3.9.3	Used by:	85
3.9.4	Substituted by:	85
3.9.5	Figure:	85
3.9.6	Schema Fragment:	85
4	Groups	86
4.1	BondChoice.model	87
4.1.1	Description:	87
4.1.2	Contents:	87
4.1.3	Used by:	87
4.1.4	Figure:	87
4.1.5	Schema Fragment:	87
4.2	BondCollateral.model	88
4.2.1	Description:	88
4.2.2	Contents:	88
4.2.3	Used by:	88
4.2.4	Figure:	88
4.2.5	Schema Fragment:	88
4.3	BondEquity.model	89
4.3.1	Description:	89
4.3.2	Contents:	89
4.3.3	Used by:	89
4.3.4	Figure:	89
4.3.5	Schema Fragment:	89
4.4	BondPriceAndYield.model	90
4.4.1	Description:	90
4.4.2	Contents:	90
4.4.3	Used by:	90
4.4.4	Figure:	90
4.4.5	Schema Fragment:	90
4.5	Settlement.model	92
4.5.1	Description:	92
4.5.2	Contents:	92
4.5.3	Used by:	92
4.5.4	Figure:	92
4.5.5	Schema Fragment:	92
4.6	SettlementAmountOrCurrency.model	93
4.6.1	Description:	93
4.6.2	Contents:	93
4.6.3	Used by:	93
4.6.4	Figure:	93
4.6.5	Schema Fragment:	93
4.7	SettlementTransferIdentifier.model	94
4.7.1	Description:	94
4.7.2	Contents:	94
4.7.3	Used by:	94
4.7.4	Figure:	94
4.7.5	Schema Fragment:	94
4.8	TradeComponentOrIdentifierOrNet.model	95
4.8.1	Description:	95

4.8.2	Contents:	95
4.8.3	Used by:	95
4.8.4	Figure:	95
4.8.5	Schema Fragment:	95
4.9	UnitContract.model	96
4.9.1	Description:	96
4.9.2	Contents:	96
4.9.3	Used by:	96
4.9.4	Figure:	96
4.9.5	Schema Fragment:	96
5	Schema listing	97

1 Global Simple Types

1.1 DeliveryMethodEnum

1.1.1 Description:

This enumeration defines the possible delivery methods for securities.

1.1.2 Contents:

Inherited element(s): (This definition restricts the content defined by the type xsd:token)

Value	Source	Description
DeliveryVersusPayment		Indicates that a securities delivery must be made against payment in a single transaction
FreeOfPayment		Indicates that a securities delivery can be made without a simultaneous cash payment in exchange.

1.1.3 Used by:

1.1.4 Derived Types:

1.1.5 Schema Fragment:

```
<xsd:simpleType name="DeliveryMethodEnum">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This enumeration defines the possible delivery methods for
      securities.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="DeliveryVersusPayment">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Indicates that a securities delivery must be made against
          payment in a single transaction
        </xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="FreeOfPayment">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Indicates that a securities delivery can be made without a
          simultaneous cash payment in exchange.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>
```

1.2 MarginTypeEnum

1.2.1 Description:

This indicator defines if a margin (also called haircut in repo and sec lending literature) applies to cash or to the financial instrument being exchanged.

1.2.2 Contents:

Inherited element(s): (This definition restricts the content defined by the type xsd:token)

Value	Source	Description
Cash		When the margin type is Cash, the margin factor is applied to the cash value of the transaction.
Instrument		When the margin type is Instrument, the margin factor is applied to the instrument value for the transaction.

1.2.3 Used by:

1.2.4 Derived Types:

1.2.5 Schema Fragment:

```
<xsd:simpleType name="MarginTypeEnum">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This indicator defines if a margin (also called haircut in repo
      and sec lending literature) applies to cash or to the financial
      instrument being exchanged.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="Cash">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          When the margin type is Cash, the margin factor is applied to
          the cash value of the transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="Instrument">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          When the margin type is Instrument, the margin factor is
          applied to the instrument value for the transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>
```

1.3 RepoDurationEnum

1.3.1 Description:

A duration code for a Repo (or Securities Lending) transaction. There are many business and market rules that are derived from the duration of the transaction.

1.3.2 Contents:

Inherited element(s): (This definition restricts the content defined by the type xsd:token)

Value	Source	Description
Overnight		Indicates that a contract is classified as overnight, meaning that there is one business day difference between the start and end date of the contract. Business rule: When the repo is overnight, the number of business days between the spot and forward value dates must be one. Forward leg must be specified.
Term		Indicates that a contract is a regular term contract, with a start date and an end date. Business rule: When the repo is 'Term', both spot and forward legs must be specified.
Open		Indicates that a contract is open ended; this means that the end date is unspecified, and will be agreed by the two parties at a later date. Business rule: When the repo is Open, the forward transaction leg must not be present.

1.3.3 Used by:

1.3.4 Derived Types:

1.3.5 Schema Fragment:

```
<xsd:simpleType name="RepoDurationEnum">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A duration code for a Repo (or Securities Lending) transaction.
      There are many business and market rules that are derived from
      the duration of the transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="Overnight">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Indicates that a contract is classified as overnight, meaning
          that there is one business day difference between the start
          and end date of the contract. Business rule: When the repo is
```

```

overnight, the number of business days between the spot and
forward value dates must be one. Forward leg must be
specified.
</xsd:documentation>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Term">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Indicates that a contract is a regular term contract, with a
      start date and an end date. Business rule: When the repo is
      'Term', both spot and forward legs must be specified.
    </xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Open">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Indicates that a contract is open ended; this means that the
      end date is unspecified, and will be agreed by the two
      parties at a later date. Business rule: When the repo is
      Open, the forward transaction leg must not be present.
    </xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>

```


2 Global Complex Types

2.1 AdjustableOffset

2.1.1 Description:

An adjustable offset can be used to specify a number of days, business or calendar, for example in a notice period.

2.1.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Offset)

- A type defining an offset used in calculating a new date relative to a reference date. Currently, the only offsets defined are expected to be expressed as either calendar or business day offsets.

Either

businessCentersReference (exactly one occurrence; of the type BusinessCentersReference) 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 (exactly one occurrence; of the type BusinessCenters)

2.1.3 Used by:

- Complex type: Repo

2.1.4 Derived Types:

2.1.5 Figure:

2.1.6 Schema Fragment:

```
<xsd:complexType name="AdjustableOffset">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An adjustable offset can be used to specify a number of days,
      business or calendar, for example in a notice period.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Offset">
      <xsd:sequence>
        <xsd:group ref="BusinessCentersOrReference.model" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.2 AtomicSettlementTransfer

2.2.1 Description:

Abstract base class for all transfer of cash or securities

2.2.2 Contents:

suppress (zero or one occurrence; of the type xsd:boolean) Indicate if this transfer should be suppressed. Absence of this flag means that the transfer should not be suppressed.

2.2.3 Used by:

- Complex type: CashTransfer
- Complex type: SecurityTransfer

2.2.4 Derived Types:

- Complex type: CashTransfer
- Complex type: SecurityTransfer

2.2.5 Figure:

2.2.6 Schema Fragment:

```
<xsd:complexType name="AtomicSettlementTransfer" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Abstract base class for all transfer of cash or securities
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="suppress" type="xsd:boolean" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Indicate if this transfer should be suppressed. Absence of
          this flag means that the transfer should not be suppressed.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID"/>
</xsd:complexType>
```

2.3 Attribution

2.3.1 Description:

An attribution must specify its type, and an amount. Any of the three amount fields below can be used (and they can all be used at the same time), as long as they are used consistently. You can express an attribution in a maximum of three different currencies (settlement, base, underlying), which are usually the same as the settlement currency on the trade, the base currency used for accounting purposes, and the underlying currency which refers to the currency of an underlying instrument used in a transaction. Note however that these are just guidelines; you can actually specify attributions in any currency that you like, as long as you are consistent. Within an Attributions structure, all attribution/settlementAmounts are expressed in the same currency, defined by the settlementCurrency field (see enclosing Attributions structure). Same holds true for base and underlying amounts.

2.3.2 Contents:

type (exactly one occurrence; of the type AttributionType) The attribution type. The cash settlement amount specified in the enclosing transfer will be broken down into several subcomponents (like a P&L explain), and the type of the breakdown is defined here. Typical values are in a scheme.

settlementAmount (zero or one occurrence; of the type xsd:decimal) An amount expressed in the settlement currency that was indicated in the enclosing Attributions structure. This is done to avoid repeating the currency for every amount when we know that attributions are expressed in a consistent way, with the same currencies.

baseAmount (zero or one occurrence; of the type xsd:decimal) An amount expressed in the base currency defined in the enclosing Attributions structure (see baseCurrency). If this optional field is present, baseCurrency must be defined in the enclosing structure.

underlyingAmount (zero or one occurrence; of the type xsd:decimal) An amount expressed in the currency of an underlyer (see underlyingCurrency). If this field is present then the underlyingCurrency field in the enclosing structure must be defined.

2.3.3 Used by:

- Complex type: Attributions

2.3.4 Derived Types:

2.3.5 Figure:

2.3.6 Schema Fragment:

```
<xsd:complexType name="Attribution">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An attribution must specify its type, and an amount. Any of the
      three amount fields below can be used (and they can all be used
      at the same time), as long as they are used consistently. You can
      express an attribution in a maximum of three different currencies
      (settlement, base, underlying), which are usually the same as the
      settlement currency on the trade, the base currency used for
      accounting purposes, and the underlying currency which refers to
      the currency of an underlying instrument used in a transaction.
      Note however that these are just guidelines; you can actually
      specify attributions in any currency that you like, as long as
      you are consistent. Within an Attributions structure, all
      attribution/settlementAmounts are expressed in the same currency,
      defined by the settlementCurrency field (see enclosing
      Attributions structure). Same holds true for base and underlying
      amounts.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="type" type="AttributionType">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The attribution type. The cash settlement amount specified in
          the enclosing transfer will be broken down into several
          subcomponents (like a P&L explain), and the type of the
          breakdown is defined here. Typical values are in a scheme.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```

</xsd:element>
<xsd:element name="settlementAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the settlement currency that was
      indicated in the enclosing Attributions structure. This is
      done to avoid repeating the currency for every amount when we
      know that attributions are expressed in a consistent way,
      with the same currencies.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="baseAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the base currency defined in the
      enclosing Attributions structure (see baseCurrency). If this
      optional field is present, baseCurrency must be defined in
      the enclosing structure.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="underlyingAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the currency of an underlyer ( see
      underlyingCurrency). If this field is present then the
      underlyingCurrency field in the enclosing structure must be
      defined.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

```

2.4 Attributions

2.4.1 Description:

A set of attributions, which are a way to break down a cash amount into several components, like the repo interest portion in the final cashflow of the repo, the clean price attribution, etc. An example could be, for a repo worth 1M on a security priced at 100 at maturity with a total interest of 10,000, 1M is attributed to the security 'dirty price', 95,400,000.00 is attributed to the security clean price, (I am making it up here); 10,000.00 is attributed to the repo interest, and 200 is attributed to a stamp tax. All attributions are monetary amounts.

2.4.2 Contents:

settlementCurrency (exactly one occurrence; of the type Currency) The currency that is used for all the attributions expressed with settlementAmount. The reason for this is to avoid repeating the currency (for example using FpML:Money) for every attribution amount in the structure. We therefore assume that attributions are expressed in a maximum of three currencies, which we specify here. The settlementCurrency is assumed to be the settlement currency of the trade in general cases.

baseCurrency (exactly one occurrence; of the type Currency) The currency that is used for all the attributions expressed with baseAmount. The baseCurrency is usually USD within the firm, but it is in fact driven by the accounting engine expectations.

underlyingCurrency (zero or one occurrence; of the type Currency) The currency that is used for all the attributions expressed with underlyingAmount. Underlying currency is the currency of issuance for the underlying instrument. So if you need to express attributions on a Repo settling in EUR but with GBP instruments, you would specify underlyingCurrency to be GBP.

attribution (one or more occurrences; of the type Attribution) The attributions go here. There is no limit on the number of attributions.

2.4.3 Used by:

- Complex type: CashTransfer

2.4.4 Derived Types:

2.4.5 Figure:

2.4.6 Schema Fragment:

```
<xsd:complexType name="Attributions">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A set of attributions, which are a way to break down a cash
      amount into several components, like the repo interest portion in
      the final cashflow of the repo, the clean price attribution, etc.
      An example could be, for a repo worth 1M on a security priced at
      100 at maturity with a total interest of 10,000, 1M is attributed
      to the security 'dirty price', 95,400,000.00 is attributed to the
      security clean price, ( I am making it up here ); 10,000.00 is
      attributed to the repo interest, and 200 is attributed to a stamp
      tax. All attributions are monetary amounts.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="settlementCurrency" type="Currency">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The currency that is used for all the attributions expressed
          with settlementAmount. The reason for this is to avoid
          repeating the currency (for example using FpML:Money) for
          every attribution amount in the structure. We therefore
          assume that attributions are expressed in a maximum of three
          currencies, which we specify here. The settlementCurrency is
          assumed to be the settlement currency of the trade in general
          cases.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="baseCurrency" type="Currency">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The currency that is used for all the attributions expressed
```

```

        with baseAmount. The baseCurrency is usually USD within the
        firm, but it is in fact driven by the accounting engine
        expectations.
    </xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="underlyingCurrency" type="Currency" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The currency that is used for all the attributions expressed
            with underlyingAmount. Underlying currency is the currency of
            issuance for the underlying instrument. So if you need to
            express attributions on a Repo settling in EUR but with GBP
            instruments, you would specify underlyingCurrency to be GBP.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="attribution" type="Attribution" maxOccurs="unbounded">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The attributions go here. There is no limit on the number of
            attributions.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

```

2.5 AttributionType

2.5.1 Description:

An attribution type. Values are defined in a coding scheme. Typical values are RepoInterest, StampTax, WithholdingTax.

2.5.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

•

2.5.3 Used by:

- Complex type: Attribution

2.5.4 Derived Types:

2.5.5 Figure:

2.5.6 Schema Fragment:

```
<xsd:complexType name="AttributionType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An attribution type. Values are defined in a coding scheme.
      Typical values are RepoInterest, StampTax, WithholdingTax.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="attributionTypeScheme" type="xsd:anyURI" default="http://www.fpml.org" />
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```


2.6 CashRepricingEvent

2.6.1 Description:

A cash repricing event. This type of event is an adjustment of the price of the underlying collateral done to reflect current market conditions. The par amount is preserved constant, which means that the collateral quantity is unchanged. It is the settlement amount that changes after a cash repricing, so a cash repricing will trigger a cash movement.

2.6.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

collateral (zero or one occurrence; of the type CollateralValuation)

combinedInterestPayout (exactly one occurrence; of the type xsd:boolean) This value should be set to true if we need to pay the accrued interest at the same time. If false, there is no payment attributed to the repo accrued interest, only cash movement linked to collateral re-valuation.

transfer (zero or one occurrence; of the type Transfer)

2.6.3 Used by:

- Element: cashRepricing

2.6.4 Derived Types:

2.6.5 Figure:

2.6.6 Schema Fragment:

```
<xsd:complexType name="CashRepricingEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A cash repricing event. This type of event is an adjustment of
      the price of the underlying collateral done to reflect current
      market conditions. The par amount is preserved constant, which
      means that the collateral quantity is unchanged. It is the
      settlement amount that changes after a cash repricing, so a cash
      repricing will trigger a cash movement.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="collateral" type="CollateralValuation" minOccurs="0"/>
        <xsd:element name="combinedInterestPayout" type="xsd:boolean">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              This value should be set to true if we need to pay the
              accrued interest at the same time. If false, there is no
              payment attributed to the repo accrued interest, only
              cash movement linked to collateral re-valuation.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="transfer" type="Transfer" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.7 CashTransfer

2.7.1 Description:

By definition, to specify a cash transfer, we need to say how much we want to transfer, who is the payer (correspondent) and who is the receiver (beneficiary). Those terms are used in the settlement instruction and allow us to define the direction of the movement.

2.7.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type AtomicSettlementTransfer)

- Abstract base class for all transfer of cash or securities

transferAmount (exactly one occurrence; of the type Money) The money to transfer.

payerPartyReference (exactly one occurrence; of the type PartyOrAccountReference) A reference to the party responsible for making the payments defined by this structure.

receiverPartyReference (exactly one occurrence; of the type PartyOrAccountReference) A reference to the party that receives the payments corresponding to this structure.

attributions (zero or one occurrence; of the type Attributions) For accounting, reporting or regulatory reasons, the transfer may have to be explained in a series of individual amounts. It may be possible for example to break down a transfer amount into constituents (gross, tax, net) or into individual amounts (interest, penalty) that would be netted at the transfer level. The attributions structure allows participants to explain their transfer amounts for better traceability. This is strictly optional.

2.7.3 Used by:

- Complex type: Transfer

2.7.4 Derived Types:

2.7.5 Figure:

2.7.6 Schema Fragment:

```
<xsd:complexType name="CashTransfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      By definition, to specify a cash transfer, we need to say how
      much we want to transfer, who is the payer ( correspondent ) and
      who is the receiver ( beneficiary ). Those terms are used in the
      settlement instruction and allow us to define the direction of
      the movement.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="AtomicSettlementTransfer">
      <xsd:sequence>
        <xsd:element name="transferAmount" type="Money">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The money to transfer.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:group ref="PayerReceiver.model">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The parties paying and receiving the money
            </xsd:documentation>
          </xsd:annotation>
        </xsd:group>
        <xsd:element name="attributions" type="Attributions" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              For accounting, reporting or regulatory reasons, the
              transfer may have to be explained in a series of
              individual amounts. It may be possible for example to
              break down a transfer amount into constituents (gross,
              tax, net) or into individual amounts (interest, penalty)
              that would be netted at the transfer level. The
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

```
        attributions structure allows participants to explain
        their transfer amounts for better traceability. This is
        strictly optional.
    </xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

2.8 CollateralSubstitutionEvent

2.8.1 Description:

A collateral substitution is an event where two parties, having previously entered into a repurchase agreement, or securities lending, agree that collateral used for the transaction be substituted with equivalent collateral. The new collateral must be deemed acceptable, in grade and quality and have an equivalent valuation compared substituted collateral at the time of the substitution.

2.8.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

previousCollateral (exactly one occurrence; of the type CollateralValuation) The collateral previously pledged in the transaction, properly valued as of the date of substitution.

newCollateral (exactly one occurrence; of the type CollateralValuation) The new collateral to use in the transaction, valued as of the date of substitution.

settlementTransfer (zero or one occurrence; of the type SettlementTransfer) The settlement transfer structure may be used by participants that want to explicitly restate who will deliver what, when and how.

2.8.3 Used by:

- Element: collateralSubstitution

2.8.4 Derived Types:

2.8.5 Figure:

2.8.6 Schema Fragment:

```
<xsd:complexType name="CollateralSubstitutionEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A collateral substitution is an event where two parties, having
      previously entered into a repurchase agreement, or securities
      lending, agree that collateral used for the transaction be
      substituted with equivalent collateral. The new collateral must
      be deemed acceptable, in grade and quality and have an equivalent
      valuation compared substituted collateral at the time of the
      substitution.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="previousCollateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The collateral previously pledged in the transaction,
              properly valued as of the date of substitution.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="newCollateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The new collateral to use in the transaction, valued as
              of the date of substitution.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="settlementTransfer" type="SettlementTransfer" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The settlement transfer structure may be used by
```

```
        participants that want to explicitly restate who will
        deliver what, when and how.
    </xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

2.9 CollateralValuation

2.9.1 Description:

This type is used in Repo trades, to specify the valuation of a specific piece of collateral in the transaction. Note that the structure must be used within a structure that has a date attached to it, because the valuations included (such as prices, accruals and inflation factors) are relative to a date. We do not want to add an href to a date because that would be too much complexity for nothing.

2.9.2 Contents:

assetReference (exactly one occurrence; of the type AssetReference) A reference to explicitly identify which asset is being valued.

2.9.3 Used by:

- Complex type: CashRepricingEvent
- Complex type: CollateralSubstitutionEvent
- Complex type: MarkToMarketEvent
- Complex type: RepoTransactionLeg

2.9.4 Derived Types:

2.9.5 Figure:

2.9.6 Schema Fragment:

```
<xsd:complexType name="CollateralValuation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This type is used in Repo trades, to specify the valuation of a
      specific piece of collateral in the transaction. Note that the
      structure must be used within a structure that has a date
      attached to it, because the valuations included (such as prices,
      accruals and inflation factors) are relative to a date. We do not
      want to add an href to a date because that would be too much
      complexity for nothing.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice>
      <xsd:group ref="BondCollateral.model">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            When the instrument being used in a transaction is a bond,
            the group above should be used to properly value the
            instrument, in terms of price, accruals and notional.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:group>
      <xsd:group ref="UnitContract.model">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            When the instrument being used in a transaction is an
            equity, or any contract traded in units, this group should
            be used to define the quantity, price and valuation of the
            instrument.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:group>
    </xsd:choice>
    <xsd:element name="assetReference" type="AssetReference">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A reference to explicitly identify which asset is being
          valued.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

2.10 CouponEvent

2.10.1 Description:

This structure is used for Buy-Sell Back trades to describe when a coupon is paid and what reinvestment rate will be applied. the amount is in the instrument currency. To be able to represent a buy/sell back with more than one collateral we use an href link to the underlying asset. This enables us to represent multiple coupons during the life of the trade, with different reinvestment rates, and possibly different instruments.

2.10.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

couponAmount (exactly one occurrence; of the type xsd:decimal) The cash value of the coupon paid (not the coupon rate). It should be equal to the coupon rate divided by frequency (2 for semi annual) times the notional of the bond.

reinvestmentRate (exactly one occurrence; of the type xsd:decimal) The reinvestment rate we will use on the coupon. Very often it is equal to the repo rate on the deal, but it does not have to. For very long term repos, the reinvestment rate will be derived from a curve.

assetReference (exactly one occurrence; of the type AssetReference)

transfer (zero or one occurrence; of the type Transfer) The transfer structure can be used to explicitly state who will pay the coupon. In buy-sell-back trades, whoever holds the bond will receive the coupon (from the bond issuer) and keep it. If the bond holder passes the coupon on to the counterparty we expect to see a transfer from bond holder to counterparty here.

2.10.3 Used by:

- Element: couponEvent

2.10.4 Derived Types:

2.10.5 Figure:

2.10.6 Schema Fragment:

```
<xsd:complexType name="CouponEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This structure is used for Buy-Sell Back trades to describe when
      a coupon is paid and what reinvestment rate will be applied. the
      amount is in the instrument currency. To be able to represent a
      buy/sell back with more than one collateral we use an href link
      to the underlying asset. This enables us to represent multiple
      coupons during the life of the trade, with different reinvestment
      rates, and possibly different instruments.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="couponAmount" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The cash value of the coupon paid (not the coupon rate).
              It should be equal to the coupon rate divided by
              frequency (2 for semi annual) times the notional of the
              bond.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="reinvestmentRate" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
```

The reinvestment rate we will use on the coupon. Very often it is equal to the repo rate on the deal, but it does not have to. For very long term repos, the reinvestment rate will be derived from a curve.

```
</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="assetReference" type="AssetReference"/>
<xsd:element name="transfer" type="Transfer" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The transfer structure can be used to explicitly state
      who will pay the coupon. In buy-sell-back trades, whoever
      holds the bond will receive the coupon (from the bond
      issuer) and keep it. If the bond holder passes the coupon
      on to the counterparty we expect to see a transfer from
      bond holder to counterparty here.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```


2.11 EventReference

2.11.1 Description:

Reference to a Trade Event.

2.11.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Reference)

- The abstract base class for all types which define intra-document pointers.

2.11.3 Used by:

- Complex type: TradeComponentIdentifier

2.11.4 Derived Types:

2.11.5 Figure:

2.11.6 Schema Fragment:

```
<xsd:complexType name="EventReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Reference to a Trade Event.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference"/>
  </xsd:complexContent>
</xsd:complexType>
```

2.12 ForwardRepoTransactionLeg

2.12.1 Description:

A transaction leg for a repo is equivalent to a single cash transaction. It is augmented here to carry some values that are of interest for the repo. Also note that the BuyerSeller model in this transaction must be the exact opposite of the one found in the spot leg.

2.12.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type RepoTransactionLeg)

- A transaction leg for a repo is equivalent to a single cash transaction.

repoInterest (zero or one occurrence; of the type xsd:decimal) The repo interest is basically the difference between the settlement amounts at spot and forward date. It is a fully figured amount, but it does not have to be specified in the message. It is not a 'Money' amount as it is implicitly expressed in the settlement currency.

2.12.3 Used by:

- Complex type: Repo

2.12.4 Derived Types:

2.12.5 Figure:

2.12.6 Schema Fragment:

```
<xsd:complexType name="ForwardRepoTransactionLeg">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A transaction leg for a repo is equivalent to a single cash
      transaction. It is augmented here to carry some values that are
      of interest for the repo. Also note that the BuyerSeller model in
      this transaction must be the exact opposite of the one found in
      the spot leg.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="RepoTransactionLeg">
      <xsd:sequence>
        <xsd:element name="repoInterest" type="xsd:decimal" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The repo interest is basically the difference between the
              settlement amounts at spot and forward date. It is a
              fully figured amount, but it does not have to be
              specified in the message. It is not a 'Money' amount as
              it is implicitly expressed in the settlement currency.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.13 InterestPayoutEvent

2.13.1 Description:

2.13.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

payment (exactly one occurrence; of the type Money) The amount to be paid. Note that we do not specify who is paying the amount. Implicitly, the party that pays the repo Interest at maturity is the only party that can do an interest payout.

transfer (zero or one occurrence; of the type Transfer) When necessary, it is possible to make the interest payout fully explicit, with parties and settlement instructions. Note that the transfer date *may* differ from the eventDate specified by the event; for example the date the payment is made can be 1 or 2 days after the interest has been calculated. Note that for an interest payout the transfer can only contain a cashTransfer as there are no security movements for an interest payout.

2.13.3 Used by:

- Element: interestPayout

2.13.4 Derived Types:

2.13.5 Figure:

2.13.6 Schema Fragment:

```
<xsd:complexType name="InterestPayoutEvent">
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="payment" type="Money">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The amount to be paid. Note that we do not specify who is
              paying the amount. Implicitly, the party that pays the
              repo Interest at maturity is the only party that can do
              an interest payout.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="transfer" type="Transfer" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              When necessary, it is possible to make the interest
              payout fully explicit, with parties and settlement
              instructions. Note that the transfer date may differ
              from the eventDate specified by the event; for example
              the date the payment is made can be 1 or 2 days after the
              interest has been calculated. Note that for an interest
              payout the transfer can only contain a cashTransfer as
              there are no security movements for an interest payout.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.14 Margin

2.14.1 Description:

Defines the margin, also called haircut in repo and securities lending terminology, that will be applied to a transaction. It is essentially a premium, expressed in percentage, to compensate collateral quality (or lack thereof) and liquidity.

2.14.2 Contents:

marginType (exactly one occurrence; of the type `MarginTypeEnum`) The type of margin being specified to apply to the transaction.

marginFactor (exactly one occurrence; of the type `xsd:decimal`) The margin is expressed as a multiplication factor (default value is 1) to reflect the quality of the collateral. Also called margin ratio as per Section 2, paragraph (z) of the TBMA/ISMA Global Master Repurchase Agreement.

2.14.3 Used by:

- Complex type: Repo

2.14.4 Derived Types:

2.14.5 Figure:

2.14.6 Schema Fragment:

```
<xsd:complexType name="Margin">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Defines the margin, also called haircut in repo and securities
      lending terminology, that will be applied to a transaction. It is
      essentially a premium, expressed in percentage, to compensate
      collateral quality (or lack thereof) and liquidity.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="marginType" type="MarginTypeEnum">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The type of margin being specified to apply to the
          transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="marginFactor" type="xsd:decimal">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The margin is expressed as a multiplication factor (default
          value is 1) to reflect the quality of the collateral. Also
          called margin ratio as per Section 2, paragraph (z) of the
          TBMA/ISMA Global Master Repurchase Agreement.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

2.15 MarkToMarketEvent

2.15.1 Description:

A mark to market event. This type of event is an adjustment of the price of the underlying collateral done to reflect current market conditions. Depending on the direction of the collateral valuation, one of the contracting parties will either pledge more/less collateral in the contract, or add/subtract cash from the contract value.

2.15.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

collateral (exactly one occurrence; of the type CollateralValuation) The updated collateral valuation for the contract. The change compared to the previously agreed valuation determines what securities or cash movement will occur.

combinedInterestPayout (exactly one occurrence; of the type xsd:boolean) Indicates that the parties agree to settle the accrued interest on the contract at the time of the repricing.

transfer (zero or one occurrence; of the type Transfer) Participants may use this structure to explicitly restate who is receiving/delivering cash and securities.

2.15.3 Used by:

- Element: markToMarketEvent

2.15.4 Derived Types:

2.15.5 Figure:

2.15.6 Schema Fragment:

```
<xsd:complexType name="MarkToMarketEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A mark to market event. This type of event is an adjustment of
      the price of the underlying collateral done to reflect current
      market conditions. Depending on the direction of the collateral
      valuation, one of the contracting parties will either pledge
      more/less collateral in the contract, or add/subtract cash from
      the contract value.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="collateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The updated collateral valuation for the contract. The
              change compared to the previously agreed valuation
              determines what securities or cash movement will occur.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="combinedInterestPayout" type="xsd:boolean">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              Indicates that the parties agree to settle the accrued
              interest on the contract at the time of the repricing.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="transfer" type="Transfer" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              Participants may use this structure to explicitly restate
```

```
        who is receiving/delivering cash and securities.  
    </xsd:documentation>  
  </xsd:annotation>  
</xsd:element>  
</xsd:sequence>  
</xsd:extension>  
</xsd:complexContent>  
</xsd:complexType>
```

2.16 MidLifeEvent

2.16.1 Description:

A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

2.16.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Event)

- A type defining the basic structure of FpML business events; it is refined by its derived types.

eventDate (exactly one occurrence; of the type IdentifiedDate) The date the event has been recorded.

2.16.3 Used by:

- Element: midLifeEvent
- Complex type: CashRepricingEvent
- Complex type: CollateralSubstitutionEvent
- Complex type: CouponEvent
- Complex type: InterestPayoutEvent
- Complex type: MarkToMarketEvent
- Complex type: RateChangeEvent
- Complex type: RateObservationEvent

2.16.4 Derived Types:

- Complex type: CashRepricingEvent
- Complex type: CollateralSubstitutionEvent
- Complex type: CouponEvent
- Complex type: InterestPayoutEvent
- Complex type: MarkToMarketEvent
- Complex type: RateChangeEvent
- Complex type: RateObservationEvent

2.16.5 Figure:

2.16.6 Schema Fragment:

```
<xsd:complexType name="MidLifeEvent" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A mid life event in a repo or securities lending transaction.
      Such mid life events are bilateral agreements between contracting
      parties to alter the financial profile of the transaction. Such
      examples include interest payments, collateral substitutions, or
      mark-to-market activity (cash or nominal repricing) to re-balance
      the valuation of cash and collateral in the transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Event">
      <xsd:sequence>
        <xsd:element name="eventDate" type="IdentifiedDate">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The date the event has been recorded.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

</xsd:complexType>

2.17 NetTradeIdentifier

2.17.1 Description:

Identification of a net trade and original trades.

2.17.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type PartyTradeIdentifier)

- A type defining one or more trade identifiers allocated to the trade by a party. A link identifier allows the trade to be associated with other related trades, e.g. trades forming part of a larger structured transaction. It is expected that for external communication of trade there will be only one tradeId sent in the document per party.

originalTradeIdentifier (one or more occurrences; of the type TradeIdentifierList) Identification of original trades.

2.17.3 Used by:

2.17.4 Derived Types:

2.17.5 Figure:

2.17.6 Schema Fragment:

```
<xsd:complexType name="NetTradeIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Identification of a net trade and original trades.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="PartyTradeIdentifier">
      <xsd:sequence>
        <xsd:element name="originalTradeIdentifier" type="TradeIdentifierList" minOccurs="2" maxOccurs="1">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              Identification of original trades.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.18 PartySettlementTransferInformation

2.18.1 Description:

Additional information about the settlement transfer. We will use this to carry information to drive internal processing.

2.18.2 Contents:

partyReference (exactly one occurrence; of the type PartyReference) Information contained in this structure is relative to the party being referred to.

processingInformation (exactly one occurrence; of the type SettlementTransferProcessingInformation) Any party specific information that may be required to properly execute a transfer. This can be used as a type substitution point to insert party specific structures.

2.18.3 Used by:

- Complex type: SettlementTransfer

2.18.4 Derived Types:

2.18.5 Figure:

2.18.6 Schema Fragment:

```
<xsd:complexType name="PartySettlementTransferInformation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Additional information about the settlement transfer. We will use
      this to carry information to drive internal processing.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="partyReference" type="PartyReference">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Information contained in this structure is relative to the
          party being referred to.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="processingInformation" type="SettlementTransferProcessingInformation">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Any party specific information that may be required to
          properly execute a transfer. This can be used as a type
          substitution point to insert party specific structures.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

2.19 RateChangeEvent

2.19.1 Description:

A rate change is a mid life event where on a given date, the repo rate to apply is changed as a result of a bilateral agreement between the two parties. There is no cash or security movement associated with a rate change. The repo structure allows you to specify repo rates as a schedule, so this type is not strictly required within the repo. We need it for consistency and to allow a discrete event message to declare a rate change on a trade.

2.19.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

rate (exactly one occurrence; of the type xsd:decimal) The new repo rate, in decimal format that will be applicable at the specified event date.

2.19.3 Used by:

- Element: rateChange

2.19.4 Derived Types:

2.19.5 Figure:

2.19.6 Schema Fragment:

```
<xsd:complexType name="RateChangeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A rate change is a mid life event where on a given date, the repo
      rate to apply is changed as a result of a bilateral agreement
      between the two parties. There is no cash or security movement
      associated with a rate change. The repo structure allows you to
      specify repo rates as a schedule, so this type is not strictly
      required within the repo. We need it for consistency and to allow
      a discrete event message to declare a rate change on a trade.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="rate" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The new repo rate, in decimal format that will be
              applicable at the specified event date.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.20 RateObservationEvent

2.20.1 Description:

A rate observation is used only when a Repo is done against an index (typically EONIA repos) and we want to record the observed rates during the lifetime of the trade. This is similar in structure to a rate change, but the application context is different. A rate observation has no cash or security movement attached, so there is no transfer structure here. Rate observations are required on floating rate repos to calculate the accrued repo interest.

2.20.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type MidLifeEvent)

- A mid life event in a repo or securities lending transaction. Such mid life events are bilateral agreements between contracting parties to alter the financial profile of the transaction. Such examples include interest payments, collateral substitutions, or mark-to-market activity (cash or nominal repricing) to re-balance the valuation of cash and collateral in the transaction.

rate (exactly one occurrence; of the type xsd:decimal) The observed rate at the specified event date that will be used to calculate the accrued interest on an index repo contract.

2.20.3 Used by:

- Element: rateObservation

2.20.4 Derived Types:

2.20.5 Figure:

2.20.6 Schema Fragment:

```
<xsd:complexType name="RateObservationEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A rate observation is used only when a Repo is done against an
      index (typically EONIA repos) and we want to record the observed
      rates during the lifetime of the trade. This is similar in
      structure to a rate change, but the application context is
      different. A rate observation has no cash or security movement
      attached, so there is no transfer structure here. Rate
      observations are required on floating rate repos to calculate the
      accrued repo interest.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="rate" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The observed rate at the specified event date that will
              be used to calculate the accrued interest on an index
              repo contract.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.21 RelativePrice

2.21.1 Description:

A type which represents Pricing relative to a Benchmark.

2.21.2 Contents:

spread (exactly one occurrence; of the type xsd:decimal) Basis Point spread over a Benchmark.

Either

equity (exactly one occurrence; of the type EquityAsset) Defines the underlying asset when it is a listed equity.

2.21.3 Used by:

2.21.4 Derived Types:

2.21.5 Figure:

2.21.6 Schema Fragment:

```
<xsd:complexType name="RelativePrice">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A type which represents Pricing relative to a Benchmark.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="spread" type="xsd:decimal">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Basis Point spread over a Benchmark.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:group ref="BondEquity.model" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The benchmark being referred to; either a bond or equity
          product.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
</xsd:complexType>
```

2.22 Repo

2.22.1 Description:

A Repo, modeled as an FpML:Product.

2.22.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Product)

- The base type which all FpML products extend.

Either

fixedRateSchedule (exactly one occurrence; of the type Schedule) The fixed repo rate. It is usually fixed for the duration of the agreement but can be changed with mid-life events (rate changes) except for buy SellBack trades.

Or

floatingRateCalculation (exactly one occurrence; of the type FloatingRateCalculation) The float index and tenor, with a spread. Use for floating rate repos. Most floatings in Europe are against EONIA.

dayCountFraction (exactly one occurrence; of the type DayCountFraction) The day count fraction.

noticePeriod (zero or one occurrence; of the type AdjustableOffset) Notice period for open ended or long dated repos in number of days.

duration (exactly one occurrence; of the type RepoDurationEnum) A duration code for the contract.

margin (exactly one occurrence; of the type Margin) The margin, or haircut, that will be applied.

spotLeg (exactly one occurrence; of the type RepoTransactionLeg) A repo contract is modelled as two purchase/repurchase transactions which are called legs. This is the spot leg, i.e. the transaction that will be executed on the settlement date of the contract.

forwardLeg (zero or one occurrence; of the type ForwardRepoTransactionLeg) The forward leg of the repo contract, i.e. the repurchase transaction.

midLifeEvent (zero or more occurrences; of the type MidLifeEvent) Abstract placeholder for repo mid life events.

Either

equity (exactly one occurrence; of the type EquityAsset) Defines the underlying asset when it is a listed equity.

settlementTransfer (zero or one occurrence; of the type SettlementTransfer) Participants may choose to explicitly specify the different financial exchanges that the product represents. This is the equivalent of the cashflow-explicit representation of an interest rate swap. In this case it is possible to give a transfer-explicit representation of a repo contract for settlement purposes.

2.22.3 Used by:

- Element: repo

2.22.4 Derived Types:

2.22.5 Figure:

2.22.6 Schema Fragment:

```
<xsd:complexType name="Repo">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A Repo, modeled as an FpML:Product.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Product">
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="fixedRateSchedule" type="Schedule">
```

```

<xsd:annotation>
  <xsd:documentation xml:lang="en">
    The fixed repo rate. It is usually fixed for the
    duration of the agreement but can be changed with
    mid-life events (rate changes) except for buy SellBack
    trades.
  </xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="floatingRateCalculation" type="FloatingRateCalculation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The float index and tenor, with a spread. Use for
      floating rate repos. Most floatings in Europe are
      against EONIA.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:choice>
<xsd:element name="dayCountFraction" type="DayCountFraction">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The day count fraction.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="noticePeriod" type="AdjustableOffset" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Notice period for open ended or long dated repos in
      number of days.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="duration" type="RepoDurationEnum">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A duration code for the contract.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="margin" type="Margin">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The margin, or haircut, that will be applied.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="spotLeg" type="RepoTransactionLeg">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A repo contract is modelled as two purchase/repurchase
      transactions which are called legs. This is the spot leg,
      i.e. the transaction that will be executed on the
      settlement date of the contract.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="forwardLeg" type="ForwardRepoTransactionLeg" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The forward leg of the repo contract, i.e. the repurchase
      transaction.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element ref="midLifeEvent" minOccurs="0" maxOccurs="unbounded">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Mid life events that occur during the lifetime of the
      Repo go here.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:group ref="BondEquity.model" maxOccurs="unbounded">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A list of the financial instruments that the repo
      contract may reference.
    </xsd:documentation>
  </xsd:annotation>
</xsd:group>
<xsd:element name="settlementTransfer" type="SettlementTransfer" minOccurs="0">

```

```
<xsd:annotation>
  <xsd:documentation xml:lang="en">
    Participants may choose to explicitly specify the
    different financial exchanges that the product
    represents. This is the equivalent of the
    cashflow-explicit representation of an interest rate
    swap. In this case it is possible to give a
    transfer-explicit representation of a repo contract for
    settlement purposes.
  </xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```


2.23 RepoLegId

2.23.1 Description:

A Repo Leg Identification.

2.23.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

•

2.23.3 Used by:

- Complex type: RepoTransactionLeg
- Complex type: TradeComponentIdentifier
- Complex type: VersionedRepoLegId

2.23.4 Derived Types:

2.23.5 Figure:

2.23.6 Schema Fragment:

```
<xsd:complexType name="RepoLegId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A Repo Leg Identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="repoLegIdScheme" type="xsd:anyURI" default="http://www.fpml.org/codi
      <xsd:attribute name="id" type="xsd:ID"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

2.24 RepoTransactionLeg

2.24.1 Description:

A transaction leg for a repo is equivalent to a single cash transaction.

2.24.2 Contents:

Either

id (one or more occurrences; of the type RepoLegId)

Or

versionedId (one or more occurrences; of the type VersionedRepoLegId)

buyerPartyReference (exactly one occurrence; of the type PartyOrTradeSideReference) 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 is the fixed rate payer.

sellerPartyReference (exactly one occurrence; of the type PartyOrTradeSideReference) 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.

settlementDate (exactly one occurrence; of the type AdjustableOrRelativeDate) Settlement Date.

collateral (zero or more occurrences; of the type CollateralValuation) Collateral valuation is used to carry the quantity and price details that are required to ensure that a repo contract is executed at fair value, with the value of the collateral matching the cash amount of the repo. Collateral is declared as optional here, with multiple cardinalities, since we can do a repo "Multi", with multiple instruments specified, or a "Cash Borrow/Loan" and "TriPartyRepo" with no collateral. In general cases, however it should be specified.

2.24.3 Used by:

- Complex type: ForwardRepoTransactionLeg
- Complex type: Repo

2.24.4 Derived Types:

- Complex type: ForwardRepoTransactionLeg

2.24.5 Figure:

2.24.6 Schema Fragment:

```
<xsd:complexType name="RepoTransactionLeg">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A transaction leg for a repo is equivalent to a single cash
      transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="id" type="RepoLegId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedRepoLegId" maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:group ref="BuyerSeller.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          References to the buyer and seller of the repo contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:group ref="Settlement.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The date and monetary amounts specified for the settlement of
          this transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:element name="collateral" type="CollateralValuation" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:annotation>
  <xsd:documentation xml:lang="en">
    Collateral valuation is used to carry the quantity and price
    details that are required to ensure that a repo contract is
    executed at fair value, with the value of the collateral
    matching the cash amount of the repo. Collateral is declared
    as optional here, with multiple cardinalities, since we can
    do a repo "Multi", with multiple instruments specified, or a
    "Cash Borrow/Loan" and "TriPartyRepo" with no collateral. In
    general cases, however it should be specified.
  </xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:ID" use="optional"/>
</xsd:complexType>
```

2.25 RepoTransactionLegReference

2.25.1 Description:

Reference to an Transaction Leg.

2.25.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Reference)

- The abstract base class for all types which define intra-document pointers.

2.25.3 Used by:

- Complex type: TradeComponentIdentifier

2.25.4 Derived Types:

2.25.5 Figure:

2.25.6 Schema Fragment:

```
<xsd:complexType name="RepoTransactionLegReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Reference to an Transaction Leg.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference" />
  </xsd:complexContent>
</xsd:complexType>
```

2.26 SecurityTransfer

2.26.1 Description:

The transfer of a security requires an identifier for the security, and a quantity.

2.26.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type AtomicSettlementTransfer)

- Abstract base class for all transfer of cash or securities

quantity (exactly one occurrence; of the type xsd:decimal) The quantity of asset being transferred

assetReference (exactly one occurrence; of the type AssetReference) The asset being transferred.

delivererPartyReference (exactly one occurrence; of the type PartyReference) Reference to the party delivering the asset.

receiverPartyReference (exactly one occurrence; of the type PartyReference) Reference to the party receiving the asset.

daylightIndicator (zero or one occurrence; of the type xsd:boolean) Standard settlement in Euroclear takes place in a batch on "value date - 1" (at 4 pm), to allow trades which are not included in this batch to be settled on value date, the daylight indicator can be used. The MT 540 instruction will contain an indicator which notifies Euroclear whether a transaction can be put forward for settlement intra-day. This is the "Daylight Indicator" and will be set on all transactions with Euroclear. However, to ensure they are included within intra-day settlement, the counterparty within Euroclear (ie, participant B) must also indicate intra-day settlement can take place.

2.26.3 Used by:

- Complex type: Transfer

2.26.4 Derived Types:

2.26.5 Figure:

2.26.6 Schema Fragment:

```
<xsd:complexType name="SecurityTransfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The transfer of a security requires an identifier for the
      security, and a quantity.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="AtomicSettlementTransfer">
      <xsd:sequence>
        <xsd:element name="quantity" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The quantity of asset being transferred
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="assetReference" type="AssetReference">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The asset being transferred.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="delivererPartyReference" type="PartyReference">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              Reference to the party delivering the asset.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="receiverPartyReference" type="PartyReference">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
```

```

        Reference to the party receiving the asset.
    </xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="daylightIndicator" type="xsd:boolean" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Standard settlement in Euroclear takes place in a batch
            on "value date - 1" (at 4 pm), to allow trades which are
            not included in this batch to be settled on value date,
            the daylight indicator can be used. The MT 540
            instruction will contain an indicator which notifies
            Euroclear whether a transaction can be put forward for
            settlement intra-day. This is the "Daylight Indicator"
            and will be set on all transactions with Euroclear.
            However, to ensure they are included within intra-day
            settlement, the counterparty within Euroclear (ie,
            participant B) must also indicate intra-day settlement
            can take place.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>

```

2.27 SettlementInstructionReference

2.27.1 Description:

Settlement Instruction Reference.

2.27.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Reference)

- The abstract base class for all types which define intra-document pointers.

2.27.3 Used by:

- Complex type: Transfer

2.27.4 Derived Types:

2.27.5 Figure:

2.27.6 Schema Fragment:

```
<xsd:complexType name="SettlementInstructionReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Instruction Reference.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference"/>
  </xsd:complexContent>
</xsd:complexType>
```

2.28 SettlementTransfer

2.28.1 Description:

This is a container to carry 'transfers', i.e. elementary transfers of cash or securities. Transfer instructions are coupled with settlement instructions that are referenced.

2.28.2 Contents:

Either

id (one or more occurrences; of the type SettlementTransferId)

Or

versionedId (one or more occurrences; of the type VersionedSettlementTransferId)

Either

type (one or more occurrences; of the type SettlementTransferType)

Or

versionedType (one or more occurrences; of the type VersionedSettlementTransferType)

transferInformation (zero or more occurrences; of the type PartySettlementTransferInformation) Transfer information which applies to all elementary transfers.

transfer (one or more occurrences; of the type Transfer) An elementary transfer. There can be as many transfers specified in this structure as required.

settlementInstruction (zero or more occurrences; of the type SettlementInstruction) Settlement instruction.

2.28.3 Used by:

- Complex type: CollateralSubstitutionEvent
- Complex type: Repo

2.28.4 Derived Types:

2.28.5 Figure:

2.28.6 Schema Fragment:

```
<xsd:complexType name="SettlementTransfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This is a container to carry 'transfers', i.e. elementary
      transfers of cash or securities. Transfer instructions are
      coupled with settlement instructions that are referenced.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:group ref="SettlementTransferIdentifier.model" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Transfer Identifier which is optional in the
          context of a trade, but should always be used in a transfer
          message.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:element name="transferInformation" type="PartySettlementTransferInformation" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Transfer information which applies to all elementary
          transfers.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="transfer" type="Transfer" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          An elementary transfer. There can be as many transfers
          specified in this structure as required.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```



```
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="settlementInstruction" type="SettlementInstruction" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Settlement instruction.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
```

2.29 SettlementTransferId

2.29.1 Description:

Settlement Transfer identification.

2.29.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

•

2.29.3 Used by:

- Complex type: VersionedSettlementTransferId

2.29.4 Derived Types:

2.29.5 Figure:

2.29.6 Schema Fragment:

```
<xsd:complexType name="SettlementTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="settlementTransferIdScheme" type="xsd:anyURI" default="http://www.fpi.com" />
      <xsd:attribute name="id" type="xsd:ID"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

2.30 SettlementTransferIdentifier

2.30.1 Description:

Settlement Transfer identification and classification.

2.30.2 Contents:

Either

id (one or more occurrences; of the type SettlementTransferId)

Or

versionedId (one or more occurrences; of the type VersionedSettlementTransferId)

Either

type (one or more occurrences; of the type SettlementTransferType)

Or

versionedType (one or more occurrences; of the type VersionedSettlementTransferType)

2.30.3 Used by:

2.30.4 Derived Types:

2.30.5 Figure:

2.30.6 Schema Fragment:

```
<xsd:complexType name="SettlementTransferIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer identification and classification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:group ref="SettlementTransferIdentifier.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          An identifier for an individual settlement transfer. This
          usually maps to a SWIFT message reference.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID" use="optional"/>
</xsd:complexType>
```

2.31 SettlementTransferProcessingInformation

2.31.1 Description:

Type representing settlement transfer processing information.

2.31.2 Contents:

owner (zero or one occurrence; of the type xsd:boolean) If present and true the publisher considers this party to be the owner of the transfer.

2.31.3 Used by:

- Complex type: PartySettlementTransferInformation

2.31.4 Derived Types:

2.31.5 Figure:

2.31.6 Schema Fragment:

```
<xsd:complexType name="SettlementTransferProcessingInformation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Type representing settlement transfer processing information.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="owner" type="xsd:boolean" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          If present and true the publisher considers this party to be
          the owner of the transfer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

2.32 SettlementTransferType

2.32.1 Description:

A settlement transfer classification.

2.32.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

-

2.32.3 Used by:

- Complex type: VersionedSettlementTransferType

2.32.4 Derived Types:

2.32.5 Figure:

2.32.6 Schema Fragment:

```
<xsd:complexType name="SettlementTransferType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A settlement transfer classification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="settlementTransferTypeScheme" type="xsd:anyURI" default="http://www.
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

2.33 StreamId

2.33.1 Description:

Stream identification.

2.33.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

•

2.33.3 Used by:

- Complex type: TradeComponentIdentifier
- Complex type: VersionedStreamId

2.33.4 Derived Types:

2.33.5 Figure:

2.33.6 Schema Fragment:

```
<xsd:complexType name="StreamId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Stream identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="streamId" type="xsd:anyURI" default="http://www.fpml.org/coding-scheme" />
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

2.34 StreamReference

2.34.1 Description:

Reference to an Stream.

2.34.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type Reference)

- The abstract base class for all types which define intra-document pointers.

2.34.3 Used by:

- Complex type: TradeComponentIdentifier

2.34.4 Derived Types:

2.34.5 Figure:

2.34.6 Schema Fragment:

```
<xsd:complexType name="StreamReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Reference to an Stream.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference"/>
  </xsd:complexContent>
</xsd:complexType>
```

2.35 TradeAndComponentIdentifier

2.35.1 Description:

Contains identification of a trade, and references to a trade component or event.

2.35.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type TradeIdentifier)

- A type defining a trade identifier issued by the indicated party.

tradeComponentIdentifier (exactly one occurrence; of the type TradeComponentIdentifier)

2.35.3 Used by:

2.35.4 Derived Types:

2.35.5 Figure:

2.35.6 Schema Fragment:

```
<xsd:complexType name="TradeAndComponentIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Contains identification of a trade, and references to a trade
      component or event.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="TradeIdentifier">
      <xsd:sequence>
        <xsd:element name="tradeComponentIdentifier" type="TradeComponentIdentifier"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```


2.36 TradeComponentIdentifier

2.36.1 Description:

Trade Component or Event identification or references.

2.36.2 Contents:

Either

repoLegReference (exactly one occurrence; of the type RepoTransactionLegReference)

Or

eventReference (exactly one occurrence; of the type EventReference)

Or

streamReference (exactly one occurrence; of the type StreamReference)

2.36.3 Used by:

- Complex type: TradeAndComponentIdentifier

2.36.4 Derived Types:

2.36.5 Figure:

2.36.6 Schema Fragment:

```
<xsd:complexType name="TradeComponentIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Trade Component or Event identification or references.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="repoLegId" type="RepoLegId"/>
      <xsd:element name="versionedRepoLegId" type="VersionedRepoLegId"/>
    </xsd:choice>
    <xsd:element name="repoLegReference" type="RepoTransactionLegReference"/>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="eventId" type="EventId"/>
      <xsd:element name="versionedEventId" type="VersionedEventId"/>
    </xsd:choice>
    <xsd:element name="eventReference" type="EventReference"/>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="streamId" type="StreamId"/>
      <xsd:element name="versionedStreamId" type="VersionedStreamId"/>
    </xsd:choice>
    <xsd:element name="streamReference" type="StreamReference"/>
  </xsd:choice>
</xsd:complexType>
```

2.37 TradelfentifierList

2.37.1 Description:

A type containing multiple tradelfentifier.

2.37.2 Contents:

tradelfentifier (one or more occurrences; of the type Tradelfentifier) A trade identifier.

2.37.3 Used by:

- Complex type: NetTradelfentifier

2.37.4 Derived Types:

2.37.5 Figure:

2.37.6 Schema Fragment:

```
<xsd:complexType name="TradeIdentifierList">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A type containing multiple tradeIdentifier.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="tradeIdentifier" type="TradeIdentifier" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A trade identifier.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

2.38 Transfer

2.38.1 Description:

A type used to represent a transfer of cash, or securities, or a simultaneous exchange of securities vs cash.

2.38.2 Contents:

Either

id (one or more occurrences; of the type TransferId)

Or

versionedId (one or more occurrences; of the type VersionedTransferId)

Either

tradeComponentIdentifier (exactly one occurrence; of the type TradeComponentIdentifier) Identify the trade component which has caused this transfer to occur within a Trade Settlement.

Or

tradeAndComponentIdentifier (one or more occurrences; of the type TradeAndComponentIdentifier) Identify the trade and component which has caused this transfer to occur within a Settlement Message.

Or

netTradeIdentifier (exactly one occurrence; of the type NetTradeIdentifier) Identify the net trade and original trades which have caused this transfer to occur within a Trade or Settlement Message.

deliveryMethod (exactly one occurrence; of the type DeliveryMethodEnum) Specify the delivery method. There is a business rule associated with this field: if deliveryMethod is DVP then you must specify a cashTransfer and a securityTransfer at the same time. It is incorrect to specify DVP and give only a cash transfer instruction.

transferDate (exactly one occurrence; of the type IdentifiedDate) The date at which the transfer should occur.

Either

securityTransfer (exactly one occurrence; of the type SecurityTransfer) A transfer of securities between two parties.

settlementInstructionReference (zero or one occurrence; of the type SettlementInstructionReference) Settlement Instruction Reference.

2.38.3 Used by:

- Complex type: CashRepricingEvent
- Complex type: CouponEvent
- Complex type: InterestPayoutEvent
- Complex type: MarkToMarketEvent
- Complex type: SettlementTransfer

2.38.4 Derived Types:

2.38.5 Figure:

2.38.6 Schema Fragment:

```
<xsd:complexType name="Transfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A type used to represent a transfer of cash, or securities, or a
      simultaneous exchange of securities vs cash.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="id" type="TransferId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedTransferId" maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

```

<xsd:group ref="TradeComponentOrIdentifierOrNet.model" minOccurs="0"/>
<xsd:element name="deliveryMethod" type="DeliveryMethodEnum">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Specify the delivery method. There is a business rule
      associated with this field: if deliveryMethod is DVP then you
      must specify a cashTransfer and a securityTransfer at the
      same time. It is incorrect to specify DVP and give only a
      cash transfer instruction.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="transferDate" type="IdentifiedDate">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The date at which the transfer should occur.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:choice>
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      You can specify either a cash transfer, or a security
      transfer, or both, but the structure below cannot be empty.
      Note the semantics of the structure: If we only have a cash
      transfer it is a pure cash transfer, mapping to a MT202 or
      MT210; if we have a security transfer only, it maps to a
      MT540 or 542 (deliver or receive free). If the structure has
      both cash and security specified it maps to MT541 or MT543
      (deliver or receive against payment). The deliveryMethod tag
      allows us to validate that the transfer is structurally
      valid.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="cashTransfer" type="CashTransfer">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A transfer of a cash amount between two parties.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="securityTransfer" type="SecurityTransfer" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A transfer of securities between two parties.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:element name="securityTransfer" type="SecurityTransfer">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        A transfer of securities between two parties.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:choice>
<xsd:element name="settlementInstructionReference" type="SettlementInstructionReference" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Instruction Reference.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:ID"/>
</xsd:complexType>

```

2.39 TransferId

2.39.1 Description:

A Transfer Identification.

2.39.2 Contents:

Inherited element(s): (This definition inherits the content defined by the type xsd:normalizedString)

•

2.39.3 Used by:

- Complex type: Transfer
- Complex type: VersionedTransferId

2.39.4 Derived Types:

2.39.5 Figure:

2.39.6 Schema Fragment:

```
<xsd:complexType name="TransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A Transfer Identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="transferIdScheme" type="xsd:anyURI" default="http://www.fpml.org/cod" />
      <xsd:attribute name="id" type="xsd:ID" />
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

2.40 VersionedEventId

2.40.1 Description:

EventId with version control elements.

2.40.2 Contents:

id (exactly one occurrence; of the type EventId)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.40.3 Used by:

- Complex type: TradeComponentIdentifier

2.40.4 Derived Types:

2.40.5 Figure:

2.40.6 Schema Fragment:

```
<xsd:complexType name="VersionedEventId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      EventId with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="EventId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```

2.41 VersionedRepoLegId

2.41.1 Description:

Repo Leg Id with version control elements.

2.41.2 Contents:

id (exactly one occurrence; of the type RepoLegId)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.41.3 Used by:

- Complex type: RepoTransactionLeg
- Complex type: TradeComponentIdentifier

2.41.4 Derived Types:

2.41.5 Figure:

2.41.6 Schema Fragment:

```
<xsd:complexType name="VersionedRepoLegId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Repo Leg Id with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="RepoLegId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```

2.42 VersionedSettlementTransferId

2.42.1 Description:

Settlement Transfer Id with version control elements.

2.42.2 Contents:

id (exactly one occurrence; of the type SettlementTransferId)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.42.3 Used by:

2.42.4 Derived Types:

2.42.5 Figure:

2.42.6 Schema Fragment:

```
<xsd:complexType name="VersionedSettlementTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer Id with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="SettlementTransferId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```


2.43 VersionedSettlementTransferType

2.43.1 Description:

SettlementTransfer Type with version control elements.

2.43.2 Contents:

type (exactly one occurrence; of the type SettlementTransferType)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.43.3 Used by:

2.43.4 Derived Types:

2.43.5 Figure:

2.43.6 Schema Fragment:

```
<xsd:complexType name="VersionedSettlementTransferType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      SettlementTransfer Type with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="type" type="SettlementTransferType"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```

2.44 VersionedStreamId

2.44.1 Description:

StreamId with version control elements.

2.44.2 Contents:

id (exactly one occurrence; of the type StreamId)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.44.3 Used by:

- Complex type: TradeComponentIdentifier

2.44.4 Derived Types:

2.44.5 Figure:

2.44.6 Schema Fragment:

```
<xsd:complexType name="VersionedStreamId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      StreamId with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="StreamId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```

2.45 VersionedTransferId

2.45.1 Description:

Transfer Id with version control elements.

2.45.2 Contents:

id (exactly one occurrence; of the type TransferId)

version (exactly one occurrence; of the type xsd:nonNegativeInteger) The version number

effectiveDate (zero or one occurrence; of the type IdentifiedDate) Optionally it is possible to specify a version effective date when a versionId is supplied.

2.45.3 Used by:

- Complex type: Transfer

2.45.4 Derived Types:

2.45.5 Figure:

2.45.6 Schema Fragment:

```
<xsd:complexType name="VersionedTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Transfer Id with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="TransferId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
```

3 *Global Elements*

3.1 cashRepricing

3.1.1 Description:

Global element representing cash repricing.

3.1.2 Contents:

Element cashRepricing is defined by the complex type CashRepricingEvent

3.1.3 Used by:

3.1.4 Substituted by:

3.1.5 Figure:

3.1.6 Schema Fragment:

```
<xsd:element name="cashRepricing" type="CashRepricingEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing cash repricing.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.2 collateralSubstitution

3.2.1 Description:

Global element representing a collateral substitution.

3.2.2 Contents:

Element collateralSubstitution is defined by the complex type CollateralSubstitutionEvent

3.2.3 Used by:

3.2.4 Substituted by:

3.2.5 Figure:

3.2.6 Schema Fragment:

```
<xsd:element name="collateralSubstitution" type="CollateralSubstitutionEvent" substitutionGroup="CollateralSubstitutionEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a collateral substitution.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.3 couponEvent

3.3.1 Description:

Global element representing a coupon event.

3.3.2 Contents:

Element couponEvent is defined by the complex type CouponEvent

3.3.3 Used by:

3.3.4 Substituted by:

3.3.5 Figure:

3.3.6 Schema Fragment:

```
<xsd:element name="couponEvent" type="CouponEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a coupon event.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.4 interestPayout

3.4.1 Description:

Global element representing an interest payout.

3.4.2 Contents:

Element interestPayout is defined by the complex type InterestPayoutEvent

3.4.3 Used by:

3.4.4 Substituted by:

3.4.5 Figure:

3.4.6 Schema Fragment:

```
<xsd:element name="interestPayout" type="InterestPayoutEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing an interest payout.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```


3.5 markToMarketEvent

3.5.1 Description:

Global element representing a nominal repricing.

3.5.2 Contents:

Element markToMarketEvent is defined by the complex type MarkToMarketEvent

3.5.3 Used by:

3.5.4 Substituted by:

3.5.5 Figure:

3.5.6 Schema Fragment:

```
<xsd:element name="markToMarketEvent" type="MarkToMarketEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a nominal repricing.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.6 midLifeEvent

3.6.1 Description:

Abstract placeholder for repo mid life events.

3.6.2 Contents:

Element midLifeEvent is defined by the complex type MidLifeEvent

3.6.3 Used by:

- Complex type: Repo

3.6.4 Substituted by:

- Element: cashRepricing
- Element: collateralSubstitution
- Element: couponEvent
- Element: interestPayout
- Element: markToMarketEvent
- Element: rateChange
- Element: rateObservation

3.6.5 Figure:

3.6.6 Schema Fragment:

```
<xsd:element name="midLifeEvent" type="MidLifeEvent" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Abstract placeholder for repo mid life events.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.7 rateChange

3.7.1 Description:

Global element representing a rate change.

3.7.2 Contents:

Element rateChange is defined by the complex type RateChangeEvent

3.7.3 Used by:

3.7.4 Substituted by:

3.7.5 Figure:

3.7.6 Schema Fragment:

```
<xsd:element name="rateChange" type="RateChangeEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a rate change.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.8 rateObservation

3.8.1 Description:

Global element representing rate observation.

3.8.2 Contents:

Element rateObservation is defined by the complex type RateObservationEvent

3.8.3 Used by:

3.8.4 Substituted by:

3.8.5 Figure:

3.8.6 Schema Fragment:

```
<xsd:element name="rateObservation" type="RateObservationEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing rate observation.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

3.9 repo

3.9.1 Description:

3.9.2 Contents:

Element repo is defined by the complex type Repo

3.9.3 Used by:

3.9.4 Substituted by:

3.9.5 Figure:

3.9.6 Schema Fragment:

```
<xsd:element name="repo" type="Repo" substitutionGroup="product"/>
```

4 Groups

4.1 BondChoice.model

4.1.1 Description:

A model group which provides choices between all bond underlyers.

4.1.2 Contents:

Either

bond (exactly one occurrence; of the type Bond) Defines the underlying asset when it is a bond.

Or

convertibleBond (exactly one occurrence; of the type ConvertibleBond) Defines the underlying asset when it is a convertible bond.

4.1.3 Used by:

4.1.4 Figure:

4.1.5 Schema Fragment:

```
<xsd:group name="BondChoice.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A model group which provides choices between all bond underlyers.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element ref="bond">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A bond instrument referenced by a contract
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element ref="convertibleBond">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A convertible bond instrument referenced by a contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
```

4.2 BondCollateral.model

4.2.1 Description:

A group which has Collateral elements.

4.2.2 Contents:

nominalAmount (exactly one occurrence; of the type Money) Collateral nominal amount.

cleanPrice (exactly one occurrence; of the type xsd:decimal) Bond clean price, expressed in percentage points, 100 is the initial value of the bond.

accruals (zero or one occurrence; of the type xsd:decimal) Accruals, relationship is clean price and accruals equals dirty price, all prices are expressed in percentage points, 100 is the initial value of the bond.

dirtyPrice (zero or one occurrence; of the type xsd:decimal) Bond dirty price, expressed in percentage points, 100 is the initial value of the bond.

relativePrice (zero or one occurrence; of the type RelativePrice) Bond price relative to a Benchmark.

yieldToMaturity (zero or one occurrence; of the type xsd:decimal) Yield to Maturity.

inflationFactor (zero or one occurrence; of the type xsd:decimal) The inflation factor is specified for inflation-linked products which require some additional elements to calculate prices correctly.

interestStartDate (zero or one occurrence; of the type AdjustableOrRelativeDate) Start date for interest calculations.

pool (zero or one occurrence; of the type AssetPool) Characterise an asset pool, such as the MBS pool on a collateralised repo.

4.2.3 Used by:

- Complex type: CollateralValuation

4.2.4 Figure:

4.2.5 Schema Fragment:

```
<xsd:group name="BondCollateral.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has Collateral elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="nominalAmount" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Collateral nominal amount.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:group ref="BondPriceAndYield.model" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Collateral price.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
</xsd:group>
```


4.3 BondEquity.model

4.3.1 Description:

A model group that allows us to specify that a repo contract can reference bond or equity instruments.

4.3.2 Contents:

Either

equity (exactly one occurrence; of the type EquityAsset) Defines the underlying asset when it is a listed equity.

4.3.3 Used by:

- Complex type: RelativePrice
- Complex type: Repo

4.3.4 Figure:

4.3.5 Schema Fragment:

```
<xsd:group name="BondEquity.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A model group that allows us to specify that a repo contract can
      reference bond or equity instruments.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Most repos are done using Bonds and Bond subclasses as
        collateral; However it is technically possible to execute a
        repo on an equity, as long as the mark to market is correctly
        done during the lifetime of the repo.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:group ref="BondChoice.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          a bond, or bond subtype referenced by a repo contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:element ref="equity">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          an equity referenced by a repo contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
```

4.4 BondPriceAndYield.model

4.4.1 Description:

A group which has either Bond Price or Yield elements.

4.4.2 Contents:

cleanPrice (exactly one occurrence; of the type xsd:decimal) Bond clean price, expressed in percentage points, 100 is the initial value of the bond.

accruals (zero or one occurrence; of the type xsd:decimal) Accruals, relationship is clean price and accruals equals dirty price, all prices are expressed in percentage points, 100 is the initial value of the bond.

dirtyPrice (zero or one occurrence; of the type xsd:decimal) Bond dirty price, expressed in percentage points, 100 is the initial value of the bond.

relativePrice (zero or one occurrence; of the type RelativePrice) Bond price relative to a Benchmark.

yieldToMaturity (zero or one occurrence; of the type xsd:decimal) Yield to Maturity.

inflationFactor (zero or one occurrence; of the type xsd:decimal) The inflation factor is specified for inflation-linked products which require some additional elements to calculate prices correctly.

interestStartDate (zero or one occurrence; of the type AdjustableOrRelativeDate) Start date for interest calculations.

pool (zero or one occurrence; of the type AssetPool) Characterise an asset pool, such as the MBS pool on a collateralised repo.

4.4.3 Used by:

4.4.4 Figure:

4.4.5 Schema Fragment:

```
<xsd:group name="BondPriceAndYield.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has either Bond Price or Yield elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="cleanPrice" type="xsd:decimal">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Bond clean price, expressed in percentage points, 100 is the
            initial value of the bond.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="accruals" type="xsd:decimal" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Accruals, relationship is clean price and accruals equals
            dirty price, all prices are expressed in percentage points,
            100 is the initial value of the bond.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="dirtyPrice" type="xsd:decimal" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Bond dirty price, expressed in percentage points, 100 is the
            initial value of the bond.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="relativePrice" type="RelativePrice" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Bond price relative to a Benchmark.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="yieldToMaturity" type="xsd:decimal" minOccurs="0">
      <xsd:annotation>
```

```

        <xsd:documentation xml:lang="en">
            Yield to Maturity.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="inflationFactor" type="xsd:decimal" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The inflation factor is specified for inflation-linked
            products which require some additional elements to calculate
            prices correctly.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="interestStartDate" type="AdjustableOrRelativeDate" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Start date for interest calculations.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="pool" type="AssetPool" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Characterise an asset pool, such as the MBS pool on a
            collateralised repo.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:group>

```

4.5 Settlement.model

4.5.1 Description:

A group which has cash settlement elements.

4.5.2 Contents:

settlementDate (exactly one occurrence; of the type AdjustableOrRelativeDate) Settlement Date.

Either

settlementAmount (exactly one occurrence; of the type Money) Settlement Amount.

Or

settlementCurrency (exactly one occurrence; of the type Currency) Settlement Currency for use where the Settlement Amount cannot be known in advance.

4.5.3 Used by:

- Complex type: RepoTransactionLeg

4.5.4 Figure:

4.5.5 Schema Fragment:

```
<xsd:group name="Settlement.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has cash settlement elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="settlementDate" type="AdjustableOrRelativeDate">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Date.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:group ref="SettlementAmountOrCurrency.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          When the exact financial amount to trade is not known, this
          structure allows participants to state the currency of the
          transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
</xsd:group>
```

4.6 SettlementAmountOrCurrency.model

4.6.1 Description:

4.6.2 Contents:

Either

settlementAmount (exactly one occurrence; of the type Money) Settlement Amount.

Or

settlementCurrency (exactly one occurrence; of the type Currency) Settlement Currency for use where the Settlement Amount cannot be known in advance.

4.6.3 Used by:

4.6.4 Figure:

4.6.5 Schema Fragment:

```
<xsd:group name="SettlementAmountOrCurrency.model">
  <xsd:choice>
    <xsd:element name="settlementAmount" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Amount.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="settlementCurrency" type="Currency">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Currency for use where the Settlement Amount
          cannot be known in advance.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
```

4.7 SettlementTransferIdentifier.model

4.7.1 Description:

Group which has settlement transfer identifier elements.

4.7.2 Contents:

Either

id (one or more occurrences; of the type SettlementTransferId)

Or

versionedId (one or more occurrences; of the type VersionedSettlementTransferId)

Either

type (one or more occurrences; of the type SettlementTransferType)

Or

versionedType (one or more occurrences; of the type VersionedSettlementTransferType)

4.7.3 Used by:

- Complex type: SettlementTransfer
- Complex type: SettlementTransferIdentifier

4.7.4 Figure:

4.7.5 Schema Fragment:

```
<xsd:group name="SettlementTransferIdentifier.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Group which has settlement transfer identifier elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="id" type="SettlementTransferId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedSettlementTransferId" maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:choice minOccurs="0">
      <xsd:element name="type" type="SettlementTransferType" maxOccurs="unbounded"/>
      <xsd:element name="versionedType" type="VersionedSettlementTransferType" maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:group>
```

4.8 TradeComponentOrIdentifierOrNet.model

4.8.1 Description:

Trade or Trade Identifier.

4.8.2 Contents:

Either

tradeComponentIdentifier (exactly one occurrence; of the type TradeComponentIdentifier) Identify the trade component which has caused this transfer to occur within a Trade Settlement.

Or

tradeAndComponentIdentifier (one or more occurrences; of the type TradeAndComponentIdentifier) Identify the trade and component which has caused this transfer to occur within a Settlement Message.

Or

netTradeIdentifier (exactly one occurrence; of the type NetTradeIdentifier) Identify the net trade and original trades which have caused this transfer to occur within a Trade or Settlement Message.

4.8.3 Used by:

- Complex type: Transfer

4.8.4 Figure:

4.8.5 Schema Fragment:

```
<xsd:group name="TradeComponentOrIdentifierOrNet.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Trade or Trade Identifier.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element name="tradeComponentIdentifier" type="TradeComponentIdentifier">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Identify the trade component which has caused this transfer
          to occur within a Trade Settlement.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="tradeAndComponentIdentifier" type="TradeAndComponentIdentifier" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Identify the trade and component which has caused this
          transfer to occur within a Settlement Message.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="netTradeIdentifier" type="NetTradeIdentifier">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Identify the net trade and original trades which have caused
          this transfer to occur within a Trade or Settlement Message.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
```

4.9 UnitContract.model

4.9.1 Description:

A group which has unit based trade elements.

4.9.2 Contents:

numberOfUnits (exactly one occurrence; of the type xsd:decimal) The number of units (index or securities).

unitPrice (exactly one occurrence; of the type Money) The price of each unit.

4.9.3 Used by:

- Complex type: CollateralValuation

4.9.4 Figure:

4.9.5 Schema Fragment:

```
<xsd:group name="UnitContract.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has unit based trade elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="numberOfUnits" type="xsd:decimal">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The number of units (index or securities).
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="unitPrice" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The price of each unit.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:group>
```


5 Schema listing

```
<xsd:schema ecore:nsPrefix="fpml" ecore:package="org.fpml" ecore:documentRoot="FpML" targetNameSpace="http://www.fpml.org/FpML-4">
  <xsd:include schemaLocation="fpml-shared-4-3.xsd"/>
  <xsd:include schemaLocation="fpml-asset-4-3.xsd"/>
  <xsd:include schemaLocation="fpml-doc-4-3.xsd"/>
  <xsd:simpleType name="DeliveryMethodEnum">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        This enumeration defines the possible delivery methods for
        securities.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="DeliveryVersusPayment">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Indicates that a securities delivery must be made against
            payment in a single transaction
          </xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="FreeOfPayment">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Indicates that a securities delivery can be made without a
            simultaneous cash payment in exchange.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="MarginTypeEnum">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        This indicator defines if a margin (also called haircut in repo
        and sec lending literature) applies to cash or to the
        financial instrument being exchanged.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="Cash">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            When the margin type is Cash, the margin factor is applied
            to the cash value of the transaction.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="Instrument">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            When the margin type is Instrument, the margin factor is
            applied to the instrument value for the transaction.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="RepoDurationEnum">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        A duration code for a Repo (or Securities Lending) transaction.
        There are many business and market rules that are derived from
        the duration of the transaction.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="Overnight">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Indicates that a contract is classified as overnight,
            meaning that there is one business day difference between
            the start and end date of the contract. Business rule: When
            the repo is overnight, the number of business days between
            the spot and forward value dates must be one. Forward leg
            must be specified.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="Term">

```

```

<xsd:annotation>
  <xsd:documentation xml:lang="en">
    Indicates that a contract is a regular term contract, with
    a start date and an end date. Business rule: When the repo
    is 'Term', both spot and forward legs must be specified.
  </xsd:documentation>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Open">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Indicates that a contract is open ended; this means that
      the end date is unspecified, and will be agreed by the two
      parties at a later date. Business rule: When the repo is
      Open, the forward transaction leg must not be present.
    </xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="AdjustableOffset">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An adjustable offset can be used to specify a number of days,
      business or calendar, for example in a notice period.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Offset">
      <xsd:sequence>
        <xsd:group ref="BusinessCentersOrReference.model" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="AtomicSettlementTransfer" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Abstract base class for all transfer of cash or securities
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="suppress" type="xsd:boolean" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Indicate if this transfer should be suppressed. Absence of
          this flag means that the transfer should not be suppressed.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID"/>
</xsd:complexType>
<xsd:complexType name="Attribution">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An attribution must specify its type, and an amount. Any of the
      three amount fields below can be used (and they can all be used
      at the same time), as long as they are used consistently. You
      can express an attribution in a maximum of three different
      currencies (settlement, base, underlying), which are usually
      the same as the settlement currency on the trade, the base
      currency used for accounting purposes, and the underlying
      currency which refers to the currency of an underlying
      instrument used in a transaction. Note however that these are
      just guidelines; you can actually specify attributions in any
      currency that you like, as long as you are consistent. Within
      an Attributions structure, all attribution/settlementAmounts
      are expressed in the same currency, defined by the
      settlementCurrency field (see enclosing Attributions
      structure). Same holds true for base and underlying amounts.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="type" type="AttributionType">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The attribution type. The cash settlement amount specified
          in the enclosing transfer will be broken down into several
          subcomponents (like a P&L explain), and the type of the
          breakdown is defined here. Typical values are in a scheme.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>

```

```

</xsd:element>
<xsd:element name="settlementAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the settlement currency that was
      indicated in the enclosing Attributions structure. This is
      done to avoid repeating the currency for every amount when
      we know that attributions are expressed in a consistent
      way, with the same currencies.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="baseAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the base currency defined in the
      enclosing Attributions structure (see baseCurrency). If
      this optional field is present, baseCurrency must be
      defined in the enclosing structure.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="underlyingAmount" type="xsd:decimal" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      An amount expressed in the currency of an underlyer ( see
      underlyingCurrency). If this field is present then the
      underlyingCurrency field in the enclosing structure must be
      defined.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Attributions">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A set of attributions, which are a way to break down a cash
      amount into several components, like the repo interest portion
      in the final cahflow of the repo, the clean price attribution,
      etc. An example could be, for a repo worth 1M on a security
      priced at 100 at maturity with a total interest of 10,000, 1M
      is attributed to the security 'dirty price', 95,400,000.00 is
      attributed to the security clean price, ( I am making it up
      here ); 10,000.00 is attributed to the repo interest, and 200
      is attributed to a stamp tax. All attributions are monetary
      amounts.
    </xsd:documentation>
  </xsd:annotation>
</xsd:sequence>
  <xsd:element name="settlementCurrency" type="Currency">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        The currency that is used for all the attributions
        expressed with settlementAmount. The reason for this is to
        avoid repeating the currency (for example using FpML:Money)
        for every attribution amount in the structure. We therefore
        assume that attributions are expressed in a maximum of
        three currencies, which we specify here. The
        settlementCurrency is assumed to be the settlement currency
        of the trade in general cases.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="baseCurrency" type="Currency">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        The currency that is used for all the attributions
        expressed with baseAmount. The baseCurrency is usually USD
        within the firm, but it is in fact driven by the accounting
        engine expectations.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="underlyingCurrency" type="Currency" minOccurs="0">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        The currency that is used for all the attributions
        expressed with underlyingAmount. Underlying currency is the
        currency of issuance for the underlying instrument. So if
        you need to express attributions on a Repo settling in EUR
        but with GBP instruments, you would specify
        underlyingCurrency to be GBP.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>

```

```

        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="attribution" type="Attribution" maxOccurs="unbounded">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The attributions go here. There is no limit on the number
            of attributions.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AttributionType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            An attribution type. Values are defined in a coding scheme.
            Typical values are RepoInterest, StampTax, WithholdingTax.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:normalizedString">
            <xsd:attribute name="attributionTypeScheme" type="xsd:anyURI" default="http://www.fpml.org/coding-scheme/AttributionType"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="CashRepricingEvent">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A cash repricing event. This type of event is an adjustment of
            the price of the underlying collateral done to reflect current
            market conditions. The par amount is preserved constant, which
            means that the collateral quantity is unchanged. It is the
            settlement amount that changes after a cash repricing, so a
            cash repricing will trigger a cash movement.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="MidLifeEvent">
            <xsd:sequence>
                <xsd:element name="collateral" type="CollateralValuation" minOccurs="0"/>
                <xsd:element name="combinedInterestPayout" type="xsd:boolean">
                    <xsd:annotation>
                        <xsd:documentation xml:lang="en">
                            This value should be set to true if we need to pay the
                            accrued interest at the same time. If false, there is
                            no payment attributed to the repo accrued interest,
                            only cash movement linked to collateral re-valuation.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="transfer" type="Transfer" minOccurs="0"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="CashTransfer">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            By definition, to specify a cash transfer, we need to say how
            much we want to transfer, who is the payer ( correspondent )
            and who is the receiver ( beneficiary ). Those terms are used
            in the settlement instruction and allow us to define the
            direction of the movement.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="AtomicSettlementTransfer">
            <xsd:sequence>
                <xsd:element name="transferAmount" type="Money">
                    <xsd:annotation>
                        <xsd:documentation xml:lang="en">
                            The money to transfer.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:group ref="PayerReceiver.model">
                    <xsd:annotation>
                        <xsd:documentation xml:lang="en">
                            The parties paying and receiving the money
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:group>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>

```

```

<xsd:element name="attributions" type="Attributions" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      For accounting, reporting or regulatory reasons, the
      transfer may have to be explained in a series of
      individual amounts. It may be possible for example to
      break down a transfer amount into constituents (gross,
      tax, net) or into individual amounts (interest,
      penalty) that would be netted at the transfer level.
      The attributions structure allows participants to
      explain their transfer amounts for better traceability.
      This is strictly optional.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="CollateralSubstitutionEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A collateral substitution is an event where two parties, having
      previously entered into a repurchase agreement, or securities
      lending, agree that collateral used for the transaction be
      substituted with equivalent collateral. The new collateral must
      be deemed acceptable, in grade and quality and have an
      equivalent valuation compared substituted collateral at the
      time of the substitution.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="previousCollateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The collateral previously pledged in the transaction,
              properly valued as of the date of substitution.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="newCollateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The new collateral to use in the transaction, valued as
              of the date of substitution.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="settlementTransfer" type="SettlementTransfer" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The settlement transfer structure may be used by
              participants that want to explicitly restate who will
              deliver what, when and how.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="CollateralValuation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This type is used in Repo trades, to specify the valuation of a
      specific piece of collateral in the transaction. Note that the
      structure must be used within a structure that has a date
      attached to it, because the valuations included (such as
      prices, accruals and inflation factors) are relative to a date.
      We do not want to add an href to a date because that would be
      too much complexity for nothing.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice>
      <xsd:group ref="BondCollateral.model">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            When the instrument being used in a transaction is a
            bond, the group above should be used to properly value
            the instrument, in terms of price, accruals and notional.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:group>
    </xsd:choice>
  </xsd:sequence>

```

```

        </xsd:documentation>
    </xsd:annotation>
</xsd:group>
<xsd:group ref="UnitContract.model">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            When the instrument being used in a transaction is an
            equity, or any contract traded in units, this group
            should be used to define the quantity, price and
            valuation of the instrument.
        </xsd:documentation>
    </xsd:annotation>
</xsd:group>
</xsd:choice>
<xsd:element name="assetReference" type="AssetReference">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A reference to explicitly identify which asset is being
            valued.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CouponEvent">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            This structure is used for Buy-Sell Back trades to describe
            when a coupon is paid and what reinvestment rate will be
            applied. the amount is in the instrument currency. To be able
            to represent a buy/sell back with more than one collateral we
            use an href link to the underlying asset. This enables us to
            represent multiple coupons during the life of the trade, with
            different reinvestment rates, and possibly different
            instruments.
        </xsd:documentation>
    </xsd:annotation>
<xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
        <xsd:sequence>
            <xsd:element name="couponAmount" type="xsd:decimal">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        The cash value of the coupon paid (not the coupon
                        rate). It should be equal to the coupon rate divided by
                        frequency (2 for semi annual) times the notional of the
                        bond.
                    </xsd:documentation>
                </xsd:annotation>
            </xsd:element>
            <xsd:element name="reinvestmentRate" type="xsd:decimal">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        The reinvestment rate we will use on the coupon. Very
                        often it is equal to the repo rate on the deal, but it
                        does not have to. For very long term repos, the
                        reinvestment rate will be derived from a curve.
                    </xsd:documentation>
                </xsd:annotation>
            </xsd:element>
            <xsd:element name="assetReference" type="AssetReference"/>
            <xsd:element name="transfer" type="Transfer" minOccurs="0">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        The transfer structure can be used to explicitly state
                        who will pay the coupon. In buy-sell-back trades,
                        whoever holds the bond will receive the coupon (from
                        the bond issuer) and keep it. If the bond holder passes
                        the coupon on to the counterparty we expect to see a
                        transfer from bond holder to counterparty here.
                    </xsd:documentation>
                </xsd:annotation>
            </xsd:element>
        </xsd:sequence>
    </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="EventReference">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Reference to a Trade Event.
        </xsd:documentation>
    </xsd:annotation>

```

```

<xsd:complexContent>
  <xsd:extension base="Reference"/>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ForwardRepoTransactionLeg">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A transaction leg for a repo is equivalent to a single cash
      transaction. It is augmented here to carry some values that are
      of interest for the repo. Also note that the BuyerSeller model
      in this transaction must be the exact opposite of the one found
      in the spot leg.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="RepoTransactionLeg">
      <xsd:sequence>
        <xsd:element name="repoInterest" type="xsd:decimal" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The repo interest is basically the difference between
              the settlement amounts at spot and forward date. It is
              a fully figured amount, but it does not have to be
              specified in the message. It is not a 'Money' amount as
              it is implicitly expressed in the settlement currency.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="InterestPayoutEvent">
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="payment" type="Money">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The amount to be paid. Note that we do not specify who
              is paying the amount. Implicitly, the party that pays
              the repo Interest at maturity is the only party that
              can do an interest payout.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="transfer" type="Transfer" minOccurs="0">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              When necessary, it is possible to make the interest
              payout fully explicit, with parties and settlement
              instructions. Note that the transfer date *may* differ
              from the eventDate specified by the event; for example
              the date the payment is made can be 1 or 2 days after
              the interest has been calculated. Note that for an
              interest payout the transfer can only contain a
              cashTransfer as there are no security movements for an
              interest payout.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="Margin">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Defines the margin, also called haircut in repo and securities
      lending terminology, that will be applied to a transaction. It
      is essentially a premium, expressed in percentage, to
      compensate collateral quality (or lack thereof) and liquidity.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="marginType" type="MarginTypeEnum">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The type of margin being specified to apply to the
          transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>

```

```

<xsd:element name="marginFactor" type="xsd:decimal">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The margin is expressed as a multiplication factor (default
      value is 1) to reflect the quality of the collateral. Also
      called margin ratio as per Section 2, paragraph (z) of the
      TBMA/ISMA Global Master Repurchase Agreement.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MidLifeEvent" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A mid life event in a repo or securities lending transaction.
      Such mid life events are bilateral agreements between
      contracting parties to alter the financial profile of the
      transaction. Such examples include interest payments,
      collateral substitutions, or mark-to-market activity (cash or
      nominal repricing) to re-balance the valuation of cash and
      collateral in the transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Event">
      <xsd:sequence>
        <xsd:element name="eventDate" type="IdentifiedDate">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The date the event has been recorded.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="NetTradeIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Identification of a net trade and original trades.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="PartyTradeIdentifier">
      <xsd:sequence>
        <xsd:element name="originalTradeIdentifier" type="TradeIdentifierList" minOccurs="2">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              Identification of original trades.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="MarkToMarketEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A mark to market event. This type of event is an adjustment of
      the price of the underlying collateral done to reflect current
      market conditions. Depending on the direction of the collateral
      valuation, one of the contracting parties will either pledge
      more/less collateral in the contract, or add/substract cash
      from the contract value.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MidLifeEvent">
      <xsd:sequence>
        <xsd:element name="collateral" type="CollateralValuation">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The updated collateral valuation for the contract. The
              change compared to the previously agreed valuation
              determines what securities or cash movement will occur.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="combinedInterestPayout" type="xsd:boolean">
          <xsd:annotation>

```



```

        <xsd:documentation xml:lang="en">
            Indicates that the parties agree to settle the accrued
            interest on the contract at the time of the repricing.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="transfer" type="Transfer" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Participants may use this structure to explicitly
            restate who is receiving/delivering cash and
            securities.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="PartySettlementTransferInformation">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Additional information about the settlement transfer. We will
            use this to carry information to drive internal processing.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="partyReference" type="PartyReference">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Information contained in this structure is relative to the
                    party being referred to.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="processingInformation" type="SettlementTransferProcessingInformation">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Any party specific information that may be required to
                    properly execute a transfer. This can be used as a type
                    substitution point to insert party specific structures.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RateChangeEvent">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A rate change is a mid life event where on a given date, the
            repo rate to apply is changed as a result of a bilateral
            agreement between the two parties. There is no cash or security
            movement associated with a rate change. The repo structure
            allows you to specify repo rates as a schedule, so this type is
            not strictly required within the repo. We need it for
            consistency and to allow a discrete event message to declare a
            rate change on a trade.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="MidLifeEvent">
            <xsd:sequence>
                <xsd:element name="rate" type="xsd:decimal">
                    <xsd:annotation>
                        <xsd:documentation xml:lang="en">
                            The new repo rate, in decimal format that will be
                            applicable at the specified event date.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="RateObservationEvent">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A rate observation is used only when a Repo is done against an
            index (typically EONIA repos) and we want to record the
            observed rates during the lifetime of the trade. This is
            similar in structure to a rate change, but the application
            context is different. A rate observation has no cash or
            security movement attached, so there is no transfer structure

```

```

        here. Rate observations are required on floating rate repos to
        calculate the accrued repo interest.
    </xsd:documentation>
</xsd:annotation>
</xsd:complexContent>
<xsd:extension base="MidLifeEvent">
    <xsd:sequence>
        <xsd:element name="rate" type="xsd:decimal">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    The observed rate at the specified event date that will
                    be used to calculate the accrued interest on an index
                    repo contract.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="RelativePrice">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A type which represents Pricing relative to a Benchmark.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="spread" type="xsd:decimal">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Basis Point spread over a Benchmark.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:group ref="BondEquity.model" maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    The benchmark being referred to; either a bond or equity
                    product.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:group>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Repo">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A Repo, modeled as an FpML:Product.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Product">
            <xsd:sequence>
                <xsd:choice>
                    <xsd:element name="fixedRateSchedule" type="Schedule">
                        <xsd:annotation>
                            <xsd:documentation xml:lang="en">
                                The fixed repo rate. It is usually fixed for the
                                duration of the agreement but can be changed with
                                mid-life events (rate changes) except for buy
                                SellBack trades.
                            </xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                    <xsd:element name="floatingRateCalculation" type="FloatingRateCalculation">
                        <xsd:annotation>
                            <xsd:documentation xml:lang="en">
                                The float index and tenor, with a spread. Use for
                                floating rate repos. Most floatings in Europe are
                                against EONIA.
                            </xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                </xsd:choice>
                <xsd:element name="dayCountFraction" type="DayCountFraction">
                    <xsd:annotation>
                        <xsd:documentation xml:lang="en">
                            The day count fraction.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="noticePeriod" type="AdjustableOffset" minOccurs="0">
                    <xsd:annotation>

```

```

        <xsd:documentation xml:lang="en">
            Notice period for open ended or long dated repos in
            number of days.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="duration" type="RepoDurationEnum">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A duration code for the contract.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="margin" type="Margin">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The margin, or haircut, that will be applied.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="spotLeg" type="RepoTransactionLeg">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A repo contract is modelled as two purchase/repurchase
            transactions which are called legs. This is the spot
            leg, i.e. the transaction that will be executed on the
            settlement date of the contract.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="forwardLeg" type="ForwardRepoTransactionLeg" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            The forward leg of the repo contract, i.e. the
            repurchase transaction.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element ref="midLifeEvent" minOccurs="0" maxOccurs="unbounded">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Mid life events that occur during the lifetime of the
            Repo go here.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:group ref="BondEquity.model" maxOccurs="unbounded">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A list of the financial instruments that the repo
            contract may reference.
        </xsd:documentation>
    </xsd:annotation>
</xsd:group>
<xsd:element name="settlementTransfer" type="SettlementTransfer" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Participants may choose to explicitly specify the
            different financial exchanges that the product
            represents. This is the equivalent of the
            cashflow-explicit representation of an interest rate
            swap. In this case it is possible to give a
            transfer-explicit representation of a repo contract for
            settlement purposes.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="RepoLegId">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A Repo Leg Identification.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:normalizedString">
            <xsd:attribute name="repoLegIdScheme" type="xsd:anyURI" default="http://www.fpml.org/co
            <xsd:attribute name="id" type="xsd:ID"/>
        </xsd:extension>
    </xsd:simpleContent>

```

```

</xsd:complexType>
<xsd:complexType name="RepoTransactionLeg">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A transaction leg for a repo is equivalent to a single cash
      transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="id" type="RepoLegId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedRepoLegId" maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:group ref="BuyerSeller.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          References to the buyer and seller of the repo contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:group ref="Settlement.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The date and monetary amounts specified for the settlement
          of this transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
    <xsd:element name="collateral" type="CollateralValuation" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Collateral valuation is used to carry the quantity and
          price details that are required to ensure that a repo
          contract is executed at fair value, with the value of the
          collateral matching the cash amount of the repo. Collateral
          is declared as optional here, with multiple cardinalities,
          since we can do a repo "Multi", with multiple instruments
          specified, or a "Cash Borrow/Loan" and "TriPartyRepo" with
          no collateral. In general cases, however it should be
          specified.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID" use="optional"/>
</xsd:complexType>
<xsd:complexType name="RepoTransactionLegReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Reference to an Transaction Leg.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference"/>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="SecurityTransfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The transfer of a security requires an identifier for the
      security, and a quantity.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="AtomicSettlementTransfer">
      <xsd:sequence>
        <xsd:element name="quantity" type="xsd:decimal">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The quantity of asset being transferred
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="assetReference" type="AssetReference">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The asset being transferred.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="delivererPartyReference" type="PartyReference">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">

```

```

        Reference to the party delivering the asset.
    </xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="receiverPartyReference" type="PartyReference">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Reference to the party receiving the asset.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element name="daylightIndicator" type="xsd:boolean" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Standard settlement in Euroclear takes place in a batch
            on "value date - 1" (at 4 pm), to allow trades which
            are not included in this batch to be settled on value
            date, the daylight indicator can be used. The MT 540
            instruction will contain an indicator which notifies
            Euroclear whether a transaction can be put forward for
            settlement intra-day. This is the "Daylight Indicator"
            and will be set on all transactions with Euroclear.
            However, to ensure they are included within intra-day
            settlement, the counterparty within Euroclear (ie,
            participant B) must also indicate intra-day settlement
            can take place.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="SettlementInstructionReference">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Settlement Instruction Reference.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Reference"/>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="SettlementTransfer">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            This is a container to carry 'transfers', i.e. elementary
            transfers of cash or securities. Transfer instructions are
            coupled with settlement instructions that are referenced.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:group ref="SettlementTransferIdentifier.model" minOccurs="0">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Settlement Transfer Identifier which is optional in the
                    context of a trade, but should always be used in a transfer
                    message.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:group>
        <xsd:element name="transferInformation" type="PartySettlementTransferInformation" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Transfer information which applies to all elementary
                    transfers.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="transfer" type="Transfer" maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    An elementary transfer. There can be as many transfers
                    specified in this structure as required.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="settlementInstruction" type="SettlementInstruction" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    Settlement instruction.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>

```

```

    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SettlementTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="settlementTransferIdScheme" type="xsd:anyURI" default="http://www.
      <xsd:attribute name="id" type="xsd:ID"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="SettlementTransferIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer identification and classification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:group ref="SettlementTransferIdentifier.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          An identifier for an individual settlement transfer. This
          usually maps to a SWIFT message reference.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID" use="optional"/>
</xsd:complexType>
<xsd:complexType name="SettlementTransferProcessingInformation">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Type representing settlement transfer processing information.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="owner" type="xsd:boolean" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          If present and true the publisher considers this party to
          be the owner of the transfer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SettlementTransferType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A settlement transfer classification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="settlementTransferTypeScheme" type="xsd:anyURI" default="http://www.
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="StreamId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Stream identification.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="xsd:normalizedString">
      <xsd:attribute name="streamId" type="xsd:anyURI" default="http://www.fpml.org/coding-s
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="StreamReference">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Reference to an Stream.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Reference"/>
  </xsd:complexContent>

```

```

</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="TradeAndComponentIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Contains identification of a trade, and references to a trade
      component or event.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="TradeIdentifier">
      <xsd:sequence>
        <xsd:element name="tradeComponentIdentifier" type="TradeComponentIdentifier"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="TradeComponentIdentifier">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Trade Component or Event identification or references.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="repoLegId" type="RepoLegId"/>
      <xsd:element name="versionedRepoLegId" type="VersionedRepoLegId"/>
    </xsd:choice>
    <xsd:element name="repoLegReference" type="RepoTransactionLegReference"/>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="eventId" type="EventId"/>
      <xsd:element name="versionedEventId" type="VersionedEventId"/>
    </xsd:choice>
    <xsd:element name="eventReference" type="EventReference"/>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="streamId" type="StreamId"/>
      <xsd:element name="versionedStreamId" type="VersionedStreamId"/>
    </xsd:choice>
    <xsd:element name="streamReference" type="StreamReference"/>
  </xsd:choice>
</xsd:complexType>
<xsd:complexType name="TradeIdentifierList">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A type containing multiple tradeIdentifier.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="tradeIdentifier" type="TradeIdentifier" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A trade identifier.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Transfer">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A type used to represent a transfer of cash, or securities, or
      a simultaneous exchange of securities vs cash.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice minOccurs="0">
      <xsd:element name="id" type="TransferId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedTransferId" maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:group ref="TradeComponentOrIdentifierOrNet.model" minOccurs="0"/>
    <xsd:element name="deliveryMethod" type="DeliveryMethodEnum">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Specify the delivery method. There is a business rule
          associated with this field: if deliveryMethod is DVP then
          you must specify a cashTransfer and a securityTransfer at
          the same time. It is incorrect to specify DVP and give only
          a cash transfer instruction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="transferDate" type="IdentifiedDate">
      <xsd:annotation>

```

```

        <xsd:documentation xml:lang="en">
            The date at which the transfer should occur.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:choice>
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            You can specify either a cash transfer, or a security
            transfer, or both, but the structure below cannot be empty.
            Note the semantics of the structure: If we only have a cash
            transfer it is a pure cash transfer, mapping to a MT202 or
            MT210; if we have a security transfer only, it maps to a
            MT540 or 542 (deliver or receive free). If the structure
            has both cash and security specified it maps to MT541 or
            MT543 (deliver or receive against payment). The
            deliveryMethod tag allows us to validate that the transfer
            is structurally valid.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="cashTransfer" type="CashTransfer">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    A transfer of a cash amount between two parties.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="securityTransfer" type="SecurityTransfer" minOccurs="0">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    A transfer of securities between two parties.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
    <xsd:element name="securityTransfer" type="SecurityTransfer">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                A transfer of securities between two parties.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:choice>
<xsd:element name="settlementInstructionReference" type="SettlementInstructionReference">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Settlement Instruction Reference.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:ID"/>
</xsd:complexType>
<xsd:complexType name="TransferId">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A Transfer Identification.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:normalizedString">
            <xsd:attribute name="transferIdScheme" type="xsd:anyURI" default="http://www.fpml.org/c" />
            <xsd:attribute name="id" type="xsd:ID" />
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="VersionedEventId">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            EventId with version control elements.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="id" type="EventId"/>
        <xsd:group ref="VersionHistory.model"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VersionedRepoLegId">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Repo Leg Id with version control elements.
        </xsd:documentation>
    </xsd:annotation>

```



```

</xsd:annotation>
<xsd:sequence>
  <xsd:element name="id" type="RepoLegId"/>
  <xsd:group ref="VersionHistory.model"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VersionedSettlementTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Settlement Transfer Id with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="SettlementTransferId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VersionedSettlementTransferType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      SettlementTransfer Type with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="type" type="SettlementTransferType"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VersionedStreamId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      StreamId with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="StreamId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VersionedTransferId">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Transfer Id with version control elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="id" type="TransferId"/>
    <xsd:group ref="VersionHistory.model"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="cashRepricing" type="CashRepricingEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing cash repricing.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="couponEvent" type="CouponEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a coupon event.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="collateralSubstitution" type="CollateralSubstitutionEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a collateral substitution.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="interestPayout" type="InterestPayoutEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing an interest payout.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="midLifeEvent" type="MidLifeEvent" abstract="true">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Abstract placeholder for repo mid life events.
    </xsd:documentation>
  </xsd:annotation>

```

```

</xsd:annotation>
</xsd:element>
<xsd:element name="markToMarketEvent" type="MarkToMarketEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a nominal repricing.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="rateChange" type="RateChangeEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing a rate change.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="rateObservation" type="RateObservationEvent" substitutionGroup="midLifeEvent">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Global element representing rate observation.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="repo" type="Repo" substitutionGroup="product"/>
<xsd:group name="BondChoice.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A model group which provides choices between all bond
      underlyers.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element ref="bond">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A bond instrument referenced by a contract
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element ref="convertibleBond">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A convertible bond instrument referenced by a contract.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
<xsd:group name="BondCollateral.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has Collateral elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="nominalAmount" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Collateral nominal amount.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:group ref="BondPriceAndYield.model" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Collateral price.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
</xsd:group>
<xsd:group name="BondEquity.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A model group that allows us to specify that a repo contract
      can reference bond or equity instruments.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Most repos are done using Bonds and Bond subclasses as
        collateral; However it is technically possible to execute a

```

```

        repo on an equity, as long as the mark to market is correctly
        done during the lifetime of the repo.
    </xsd:documentation>
</xsd:annotation>
<xsd:group ref="BondChoice.model">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            a bond, or bond subtype referenced by a repo contract.
        </xsd:documentation>
    </xsd:annotation>
</xsd:group>
<xsd:element ref="equity">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            an equity referenced by a repo contract.
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:choice>
</xsd:group>
<xsd:group name="BondPriceAndYield.model">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A group which has either Bond Price or Yield elements.
        </xsd:documentation>
    </xsd:annotation>
</xsd:sequence>
    <xsd:element name="cleanPrice" type="xsd:decimal">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Bond clean price, expressed in percentage points, 100 is
                the initial value of the bond.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="accruals" type="xsd:decimal" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Accruals, relationship is clean price and accruals equals
                dirty price, all prices are expressed in percentage points,
                100 is the initial value of the bond.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="dirtyPrice" type="xsd:decimal" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Bond dirty price, expressed in percentage points, 100 is
                the initial value of the bond.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="relativePrice" type="RelativePrice" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Bond price relative to a Benchmark.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="yieldToMaturity" type="xsd:decimal" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Yield to Maturity.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="inflationFactor" type="xsd:decimal" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                The inflation factor is specified for inflation-linked
                products which require some additional elements to
                calculate prices correctly.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="interestStartDate" type="AdjustableOrRelativeDate" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                Start date for interest calculations.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="pool" type="AssetPool" minOccurs="0">

```

```

        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Characterise an asset pool, such as the MBS pool on a
            collateralised repo.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:group>
<xsd:group name="Settlement.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has cash settlement elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="settlementDate" type="AdjustableOrRelativeDate">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Date.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:group ref="SettlementAmountOrCurrency.model">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          When the exact financial amount to trade is not known, this
          structure allows participants to state the currency of the
          transaction.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:group>
  </xsd:sequence>
</xsd:group>
<xsd:group name="SettlementAmountOrCurrency.model">
  <xsd:choice>
    <xsd:element name="settlementAmount" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Amount.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="settlementCurrency" type="Currency">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Settlement Currency for use where the Settlement Amount
          cannot be known in advance.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:group>
<xsd:group name="SettlementTransferIdentifier.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Group which has settlement transfer identifier elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="id" type="SettlementTransferId" maxOccurs="unbounded"/>
      <xsd:element name="versionedId" type="VersionedSettlementTransferId" maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:choice minOccurs="0">
      <xsd:element name="type" type="SettlementTransferType" maxOccurs="unbounded"/>
      <xsd:element name="versionedType" type="VersionedSettlementTransferType" maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:group>
<xsd:group name="TradeComponentOrIdentifierOrNet.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Trade or Trade Identifier.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element name="tradeComponentIdentifier" type="TradeComponentIdentifier">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          Identify the trade component which has caused this transfer
          to occur within a Trade Settlement.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>

```

```

    </xsd:annotation>
  </xsd:element>
  <xsd:element name="tradeAndComponentIdentifier" type="TradeAndComponentIdentifier" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Identify the trade and component which has caused this
        transfer to occur within a Settlement Message.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="netTradeIdentifier" type="NetTradeIdentifier">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Identify the net trade and original trades which have
        caused this transfer to occur within a Trade or Settlement
        Message.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:choice>
</xsd:group>
<xsd:group name="UnitContract.model">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A group which has unit based trade elements.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="numberOfUnits" type="xsd:decimal">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The number of units (index or securities).
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="unitPrice" type="Money">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The price of each unit.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:group>
</xsd:schema>

```