kdpw_ CCP

part of kdpw_ group

Message structure:

OTC notification - automaticTerminationSummary (otcd.ntf.001.01)

Document date: 2016-03-09

Description:
OTC notification - automaticTerminationSummary

Structure:

No.	or	XML Tag	Name	Multiplicity	Type
1		KDPWDocument	KDPW system message	[1..1]	KDPWDocument
1.1		otcd.ntf.001.01	OTC notification - automaticTerminationSummary	[1..n]	otcd.ntf.001.01
1.1.1		GnlInf	General information	[1..1]	GeneralInformation
1.1.1.1		SndrMsgRef	Sender message identifier	[1..1]	Max16Text
1.1.1.2		FuncOfMsg	Function of message	[1..1]	FunctionOfMessage
1.1.1.3		CreDtTm	Message creation date	[0..1]	DateAndDateTimeChoice
1.1.1.3.1	{ lub	Dt	Date	[1..1]	ISODate
1.1.1.3.2	lub }	DtTm	Date and time	[1..1]	ISODateTime
1.1.1.4		SeqNb	Notification sequential number	[1..1]	NumberLong
1.1.1.5		NtfTp	Notification type	[1..1]	Max35Text
1.1.2		MsgData	OTC instruction details	[0..1]	MsgData
1.1.2.1		contents		[1..1]	contents
1.1.2.1.1		content		[1..n]	automaticTerminationSummary
1.1.2.1.1.1		canceledTrades	The list of trades that were cancelled as part of the Automatic Termination process	[0..1]	
1.1.2.1.1.1.1		tradeDetail	A TradeDetail record for every cancelled trade	[0..n]	tradeDetail
1.1.2.1.1.1.1.1		participant	The Clearing Member identifier for the trade Format: Alphanumeric String, length 4	[1..1]	xs:string
1.1.2.1.1.1.1.2		clientId	The Participant identifier for the non-clearing member client of the trade. This field will not be populated for trades on the house accounts of a clearing member Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.1.1.3		counterpartyId	The original counterparty clearing member of the trade Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.1.1.4		counterpartyClientId	The Participant identifier for a non-clearing member counterparty client of the trade. This field will not be populated for trades where the counterparty account is a house account for the counterparty clearing member Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.1.1.5		defaultAccountApplied	If true this indicates the Participant record has a defaultAccount recorded, and this default account was used for the trade. The default account is used when the original account id was missing or could not be identified	[1..1]	xs:boolean
1.1.2.1.1.1.1.6		externalDealId	The external identifier of the original deal for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.1.1.7		externalId	The external identifier for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.1.1.8		internalDealId	The KDPW internal identifier of the original deal for the trade Format: Numeric String	[0..1]	xs:string
1.1.2.1.1.1.1.9		positionAccountExternalId	The external identifier of the PA account for the trade Format: Alphanumeric String	[1..1]	xs:string
1.1.2.1.1.1.1.10		positionAccountInternalId	The KDPW internal identifier for the PA of the trade Format: “PA-“ + participant + “-“ + externalAccountId E.g. PA-CM01-ACCT1	[1..1]	xs:string
1.1.2.1.1.1.1.11		reason	If the trade is rejected the reason for the rejection, may also contain warnings such informative limit breaches that are raised despite the trade being ACCEPTED. Format: Free text string	[0..1]	xs:string
1.1.2.1.1.1.1.12		source	The source matching system for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.1.1.13		status	The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been reject due to Formal trade validation, beyond the	[1..1]	xs:string

No.	or	XML Tag	Name	Multiplicity	Type
			maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction). • ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record		
1.1.2.1.1.1.1.14		trade	The internal representation of the trade	[0..1]	trade
1.1.2.1.1.1.1.14.1		account	The PA account the trade is booked against. The Account is for the external party entering into the trade	[1..1]	ref
1.1.2.1.1.1.1.14.1.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.1.1.14.1.2		revisionNumber		[0..1]	xs:long
1.1.2.1.1.1.1.14.2		analysisTag	An optional name/value pair. This can be used to provide additional information for reporting purposes. NOT USED.	[0..1]	tag
1.1.2.1.1.1.1.14.3		buySell	Specifies whether a trade is a Buy or a Sell from the perspective of the external party.	[1..1]	buySell
1.1.2.1.1.1.1.14.4		deal	The id of the Deal that this Trade is part of	[0..1]	ref
1.1.2.1.1.1.1.14.4.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.1.1.14.4.2		revisionNumber		[0..1]	xs:long
1.1.2.1.1.1.1.14.5		externalTradeDescriptor	Not Used	[0..1]	externalTradeDescriptor
1.1.2.1.1.1.1.14.6		externalTradeIdentifier	The External Trade Identifier for this Trade. This is set to the trade id used in an external matching platform where appropriate	[0..1]	xs:string
1.1.2.1.1.1.1.14.7		features		[0..1]	
1.1.2.1.1.1.1.14.7.1		feature		[0..n]	xs:string
1.1.2.1.1.1.1.14.8		lastUpdated	The timestamp when the trade was last updated	[0..1]	kasmDateTime
1.1.2.1.1.1.1.14.9		originalTradeId		[0..1]	xs:string
1.1.2.1.1.1.1.14.10		product	The underlying product for the trade. The product contains the specific economic details for the trade. The required instance of OtcProduct depends on the type of trade. The following instances are supported: • Fra • Swap • Repo	[1..1]	kasmType
1.1.2.1.1.1.1.14.11		tradeDate	The date when the trade was entered into	[0..1]	kasmDate
1.1.2.1.1.1.1.14.12		tradeState	The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been rejected due to Formal trade validation, beyond the maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction). • ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record	[0..1]	xs:string
1.1.2.1.1.1.1.14.13		trader	Not Used	[0..1]	ref
1.1.2.1.1.1.1.14.13.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.1.1.14.13.2		revisionNumber		[0..1]	xs:long

No.	or	XML Tag	Name	Multiplicity	Type
1.1.2.1.1.2		participant	The Clearing Member identifier. Used in conjunction with the External ID to uniquely identify the specific PA account within Sentinel Format: Alphanumeric String, length 4	[1..1]	xs:string
1.1.2.1.1.3		externalAccountId	The External Account identifier used by the Participant Used in conjunction with the Clearing Member ID to uniquely identify the specific PA account within Sentinel Format: Alphanumeric String	[1..1]	xs:string
1.1.2.1.1.4		internalAccountId	The internal account identifier used within Sentinel Format: “PA-“ + participant + “-“ + externalAccountId E.g. PA-CM01-ACCT1	[1..1]	xs:string
1.1.2.1.1.5		updatedTrades	The list of trades that were amended as part of the Automatic Termination process	[0..1]	
1.1.2.1.1.5.1		tradeDetail	A TradeDetail record for every updated trade	[0..n]	tradeDetail
1.1.2.1.1.5.1.1		participant	The Clearing Member identifier for the trade Format: Alphanumeric String, length 4	[1..1]	xs:string
1.1.2.1.1.5.1.2		clientId	The Participant identifier for the non-clearing member client of the trade. This field will not be populated for trades on the house accounts of a clearing member Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.5.1.3		counterpartyId	The original counterparty clearing member of the trade Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.5.1.4		counterpartyClientId	The Participant identifier for a non-clearing member counterparty client of the trade. This field will not be populated for trades where the counterparty account is a house account for the counterparty clearing member Format: Alphanumeric String, length 4	[0..1]	xs:string
1.1.2.1.1.5.1.5		defaultAccountApplied	If true this indicates the Participant record has a defaultAccount recorded, and this default account was used for the trade. The default account is used when the original account id was missing or could not be identified	[1..1]	xs:boolean
1.1.2.1.1.5.1.6		externalDealId	The external identifier of the original deal for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.5.1.7		externalId	The external identifier for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.5.1.8		internalDealId	The KDPW internal identifier of the original deal for the trade Format: Numeric String	[0..1]	xs:string
1.1.2.1.1.5.1.9		positionAccountExternalId	The external identifier of the PA account for the trade Format: Alphanumeric String	[1..1]	xs:string
1.1.2.1.1.5.1.10		positionAccountInternalId	The KDPW internal identifier for the PA of the trade Format: “PA-“ + participant + “-“ + externalAccountId E.g. PA-CM01-ACCT1	[1..1]	xs:string
1.1.2.1.1.5.1.11		reason	If the trade is rejected the reason for the rejection, may also contain warnings such informative limit breaches that are raised despite the trade being ACCEPTED. Format: Free text string	[0..1]	xs:string
1.1.2.1.1.5.1.12		source	The source matching system for the trade Format: Alphanumeric String	[0..1]	xs:string
1.1.2.1.1.5.1.13		status	The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been reject due to Formal trade validation, beyond the maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction).	[1..1]	xs:string

No.	or	XML Tag	Name	Multiplicity	Type
			• ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record		
1.1.2.1.1.5.1.14		trade	The internal representation of the trade	[0..1]	trade
1.1.2.1.1.5.1.14.1		account	The PA account the trade is booked against. The Account is for the external party entering into the trade	[1..1]	ref
1.1.2.1.1.5.1.14.1.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.5.1.14.1.2		revisionNumber		[0..1]	xs:long
1.1.2.1.1.5.1.14.2		analysisTag	An optional name/value pair. This can be used to provide additional information for reporting purposes. NOT USED.	[0..1]	tag
1.1.2.1.1.5.1.14.3		buySell	Specifies whether a trade is a Buy or a Sell from the perspective of the external party.	[1..1]	buySell
1.1.2.1.1.5.1.14.4		deal	The id of the Deal that this Trade is part of	[0..1]	ref
1.1.2.1.1.5.1.14.4.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.5.1.14.4.2		revisionNumber		[0..1]	xs:long
1.1.2.1.1.5.1.14.5		externalTradeDescriptor	Not Used	[0..1]	externalTradeDescriptor
1.1.2.1.1.5.1.14.6		externalTradeIdentifier	The External Trade Identifier for this Trade. This is set to the trade id used in an external matching platform where appropriate	[0..1]	xs:string
1.1.2.1.1.5.1.14.7		features		[0..1]	
1.1.2.1.1.5.1.14.7.1		feature		[0..n]	xs:string
1.1.2.1.1.5.1.14.8		lastUpdated	The timestamp when the trade was last updated	[0..1]	kasmDateTime
1.1.2.1.1.5.1.14.9		originalTradeId		[0..1]	xs:string
1.1.2.1.1.5.1.14.10		product	The underlying product for the trade. The product contains the specific economic details for the trade. The required instance of OtcProduct depends on the type of trade. The following instances are supported: • Fra • Swap • Repo	[1..1]	kasmType
1.1.2.1.1.5.1.14.11		tradeDate	The date when the trade was entered into	[0..1]	kasmDate
1.1.2.1.1.5.1.14.12		tradeState	The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been rejected due to Formal trade validation, beyond the maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction). • ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record	[0..1]	xs:string
1.1.2.1.1.5.1.14.13		trader	Not Used	[0..1]	ref
1.1.2.1.1.5.1.14.13.1		cacheNameOverride		[0..1]	xs:string
1.1.2.1.1.5.1.14.13.2		revisionNumber		[0..1]	xs:long

Message component elements:

KDPWDocument - KDPW system message (element)

Description KDPW system message
Type KDPWDocument
Attributes

GnlInf - General information (element)

Description General information
Type GeneralInformation
Attributes

MsgData - OTC instruction details (element)

Description OTC instruction details
Type MsgData
Attributes minOccurs=0

otcd.ntf.001.01 - OTC notification - automaticTerminationSummary (element)

Description OTC notification - automaticTerminationSummary
Type otcd.ntf.001.01
Attributes maxOccurs=unbounded

SndrMsgRef - Sender message identifier (element)

Description Sender message identifier
Type Max16Text
Attributes

FuncOfMsg - Function of message (element)

Description Function of message
Type FunctionOfMessage
Attributes

CreDtTm - Message creation date (element)

Description Message creation date
Type DateAndDateTimeChoice
Attributes minOccurs=0

SeqNb - Notification sequential number (element)

Description Notification sequential number
Type NumberLong
Attributes

NtfTp - Notification type (element)

Description Notification type
Type Max35Text
Attributes

Dt - Date (element)

Description Date
Type ISODate
Attributes

DtTm - Date and time (element)

Description Date and time
Type ISODateTime
Attributes

contents - (element)

Description
Type contents
Attributes

content - (element)

Description
Type automaticTerminationSummary
Attributes maxOccurs=unbounded

canceledTrades - (element)

Description The list of trades that were cancelled as part of the Automatic Termination process
Type
Attributes minOccurs=0

tradeDetail - (element)

Description A TradeDetail record for every cancelled trade
Type tradeDetail
Attributes minOccurs=0
maxOccurs=unbounded

participant - (element)

Description The Clearing Member identifier. Used in conjunction with the External ID to uniquely identify the specific PA account within Sentinel Format: Alphanumeric String, length 4
Type xs:string
Attributes

externalAccountId - (element)

Description The External Account identifier used by the Participant Used in conjunction with the Clearing Member ID to uniquely identify the specific PA account within Sentinel Format: Alphanumeric String
Type xs:string
Attributes

internalAccountId - (element)

Description The internal account identifier used within Sentinel Format: “PA-“ + participant + “-“ + externalAccountId E.g. PA-CM01-ACCT1
Type xs:string
Attributes

updatedTrades - (element)

Description The list of trades that were amended as part of the Automatic Termination process
Type
Attributes minOccurs=0

tradeDetail - (element)

Description A TradeDetail record for every updated trade
Type tradeDetail
Attributes minOccurs=0
maxOccurs=unbounded

participant - (element)

Description The Clearing Member identifier for the trade Format: Alphanumeric String, length 4
Type xs:string
Attributes

clientId - (element)

Description The Participant identifier for the non-clearing member client of the trade. This field will not be populated for trades on the house accounts of a clearing member Format: Alphanumeric String, length 4
Type xs:string
Attributes minOccurs=0

counterpartyId - (element)

Description The original counterparty clearing member of the trade Format: Alphanumeric String, length 4
Type xs:string
Attributes minOccurs=0

counterpartyClientId - (element)

Description The Participant identifier for a non-clearing member counterparty client of the trade. This field will not be populated for trades where the counterparty account is a house account for the counterparty clearing member Format: Alphanumeric String, length 4

Type xs:string

Attributes minOccurs=0

defaultAccountApplied - (element)

Description If true this indicates the Participant record has a defaultAccount recorded, and this default account was used for the trade. The default account is used when the original account id was missing or could not be identified

Type xs:boolean

Attributes

externalDealId - (element)

Description The external identifier of the original deal for the trade Format: Alphanumeric String

Type xs:string

Attributes minOccurs=0

externalId - (element)

Description The external identifier for the trade Format: Alphanumeric String

Type xs:string

Attributes minOccurs=0

internalDealId - (element)

Description The KDPW internal identifier of the original deal for the trade Format: Numeric String

Type xs:string

Attributes minOccurs=0

positionAccountExternalId - (element)

Description The external identifier of the PA account for the trade Format: Alphanumeric String

Type xs:string

Attributes

positionAccountInternalId - (element)

Description The KDPW internal identifier for the PA of the trade Format: “PA-“ + participant + “-“ + externalAccountId E.g. PA-CM01-ACCT1

Type xs:string

Attributes

reason - (element)

Description If the trade is rejected the reason for the rejection, may also contain warnings such informative limit breaches that are raised despite the trade being ACCEPTED. Format: Free text string
Type xs:string
Attributes minOccurs=0

source - (element)

Description The source matching system for the trade Format: Alphanumeric String
Type xs:string
Attributes minOccurs=0

status - (element)

Description The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been reject due to Formal trade validation, beyond the maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction). • ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record
Type xs:string
Attributes

trade - (element)

Description The internal representation of the trade
Type trade
Attributes minOccurs=0

account - (element)

Description The PA account the trade is booked against. The Account is for the external party entering into the trade
Type ref
Attributes

analysisTag - (element)

Description An optional name/value pair. This can be used to provide additional information for reporting purposes. NOT USED.
Type tag
Attributes minOccurs=0

buySell - (element)

Description Specifies whether a trade is a Buy or a Sell from the perspective of the external party.
Type buySell
Attributes

deal - (element)

Description The id of the Deal that this Trade is part of
Type ref
Attributes minOccurs=0

externalTradeDescriptor - (element)

Description Not Used
Type externalTradeDescriptor
Attributes minOccurs=0

externalTradeIdentifier - (element)

Description The External Trade Identifier for this Trade. This is set to the trade id used in an external matching platform where appropriate
Type xs:string
Attributes minOccurs=0

features - (element)

Description
Type
Attributes minOccurs=0

feature - (element)

Description
Type xs:string
Attributes minOccurs=0
maxOccurs=unbounded

lastUpdated - (element)

Description The timestamp when the trade was last updated
Type kasmDateTime
Attributes minOccurs=0

originalTradeId - (element)

Description
Type xs:string
Attributes minOccurs=0

product - (element)

Description The underlying product for the trade. The product contains the specific economic details for the trade. The required instance of OtcProduct depends on the type of trade. The following instances are supported: • Fra • Swap • Repo

Type kasmType

Attributes

tradeDate - (element)

Description The date when the trade was entered into

Type kasmDate

Attributes minOccurs=0

tradeState - (element)

Description The current status of the trade • QUEUED (Set if the trade was submitted while the system was in a Closed state). • ACCEPTED (Normally accepted trade status once it has passed all limit checks). • REJECTED (The trade has been rejected due to Formal trade validation, beyond the maximum number of outdated days or the Participant is suspended or in default). • PENDING (A trade that has failed limit checks and is being held in a Pending state for backloading). • BATCHED (Set if the trade was pre-flagged to be grouped into a batch and processed atomically, e.g. trades as a result of winning an Auction). • ACCEPTED_CONDITIONALLY (A trade that was accepted but indicates that there is outstanding Collateral to be posted). • PORTED – The trades account has been changed through a porting process by the CCP • DEFAULT An undefined state, usually seen in prospective trades such as those presented in an Auction Detail record

Type xs:string

Attributes minOccurs=0

trader - (element)

Description Not Used

Type ref

Attributes minOccurs=0

cacheNameOverride - (element)

Description

Type xs:string

Attributes minOccurs=0

revisionNumber - (element)

Description

Type xs:long

Attributes minOccurs=0

agreement - (element)

Description The agreement ID, for example an ISDA agreement ID.
Type xs:string
Attributes minOccurs=0

agreementType - (element)

Description The type of agreement, for example an ISDA.
Type xs:string
Attributes minOccurs=0

calendar - (element)

Description The Calendar to use to determine if the 'Spot' or Effective date of an OTC Product is a business day
Type ref
Attributes minOccurs=0

currency - (element)

Description The currency the Fra is denominated in
Type ref
Attributes

dayCountFraction - (element)

Description The DayCount convention for the Fra fixed rate. The DayCount convention specifies how the fixed rate should be treated when calculating interest amounts
Type dayCountFraction
Attributes minOccurs=0

effectiveBusinessDayConvention - (element)

Description The Business day convention used to adjust the Fra product effective date.
Type businessDayConvention
Attributes minOccurs=0

effectiveCalendar - (element)

Description The Reference to the Calendar used to adjust the Effective date for the Fra
Type ref
Attributes minOccurs=0

effectiveDate - (element)

Description The unadjusted Effective (also known as Spot) Date of the FowardRateAgreement. This is both the Payment date and the start date for accrual of interest for the ForwardRateAgreement
Type kasmDate
Attributes

expiryBusinessDayConvention - (element)

Description The Business day convention used to adjust the Fra product expiry date.
Type businessDayConvention
Attributes minOccurs=0

expiryCalendar - (element)

Description The Reference to the Calendar used to adjust the Expiry date for the Fra
Type ref
Attributes minOccurs=0

expiryDate - (element)

Description The Expiry (also known as Accrual End) Date of the Fra product. The expiry date is the unadjusted end date of the interest accrual period. If not set explicitly this date is derived from the Fra Effective date and Tenor
Type kasmDate
Attributes minOccurs=0

fees - (element)

Description The list of Fee payments attached to the OTC Product. None, one or many fee payments are supported for an OTC Product
Type
Attributes minOccurs=0

fee - (element)

Description Details of a fee payment for an OTC Product
Type fee
Attributes minOccurs=0
maxOccurs=unbounded

floatingTermStructure - (element)

Description The term structure that is used to derive the forward floating rate for the Fra. Typically this is not specified and is derived from configuration data based upon the currency and refix rate. It will override any configuration data termStructureConfig settings if specified.
Type ref
Attributes minOccurs=0

fraTenor - (element)

Description The Tenor interval of the Fra product. The tenor is the time interval between the Fra Effective Date and the Fra Expiry Date. If specified and ExpiryDate is unspecified, the Expiry Date is derived from the Effective Date and fraTenor rather than being set explicitly. One of either the fraTenor or the expiryDate must be set. expiryDate has precedence over the fraTenor

Type interval

Attributes minOccurs=0

issuer - (element)

Description

Type xs:string

Attributes minOccurs=0

notional - (element)

Description The notional amount of the Fra

Type xs:double

Attributes

productSubType - (element)

Description The sub-type of the product, if a further level of definition is needed on the type of a product in addition to the productType field. This can be used for more concise ProductConfig mappings for example when adjustments to the base product types logic or calculation is required.

Type xs:string

Attributes minOccurs=0

productType - (element)

Description The type of Product. This is set automatically by the underlying product class. It should be overridden for specific product types (indicated by Set manually in the Validation column. The supported productType values are shown in the Validation column

Type xs:string

Attributes

refixDate - (element)

Description The refixDate when the Fra forward rate is sampled. If not set this date is derived from the effectiveDate and refixRate convention

Type kasmDate

Attributes minOccurs=0

interestRateDef - (element)

Description The id of the refix rate that is used for deriving the floating refix rate for the Fra
Type ref
Attributes

settlementCashflows - (element)

Description Structure containing a list of SettlementFlows.
Type settlementCashflows
Attributes minOccurs=0

spotBdc - (element)

Description The BusinessDayConvention used to adjust the 'Spot' or Effective date of the OTC Product.
Type businessDayConvention
Attributes minOccurs=0

termStructureConfig - (element)

Description The term structure that is used to value the Fra. The specified term structure is used to discount the future Fra payment amount to determine its current Net Present Value (NPV). Typically this is not specified and is derived from configuration data based upon the currency. It will override any configuration data termStructureConfig settings if specified.
Type ref
Attributes minOccurs=0

terminationDate - (element)

Description Deprecated – this is no longer required. The termination date is derived from the product attributes
Type kasmDate
Attributes minOccurs=0

tradedRate - (element)

Description The fixed rate for the Fra product. This is the fixed rate that is being paid/received by the Fra buyer/seller
Type xs:double
Attributes

amount - (element)

Description Fee payment amount If the payReceive element is not specified: A Positive value indicates the Fee is received by the Buyer. A Negative value indicates the Fee is paid by the Buyer.
Type xs:double
Attributes

businessDayConvention - (element)

Description If not provided MODFOLLOWING will be used
Type businessDayConvention
Attributes minOccurs=0

calendar - (element)

Description If not provided the calendar from the Discount Curve will be used
Type ref
Attributes minOccurs=0

currency - (element)

Description The currency the Fee is denominated in
Type ref
Attributes

payReceive - (element)

Description Indicates the payment direction for the Buyer of the trade. If the trade is of direction BUY • PAY – The party will pay the fee • RECEIVE – The party will receive the fee If the trade is of direction SELL • PAY – The party will receive the fee • RECEIVE – The party will pay the fee
Type payReceive
Attributes minOccurs=0

paymentDate - (element)

Description Unadjusted Fee payment date
Type kasmDate
Attributes

type - (element)

Description A classification of the fee type, for information purposes only Will default to UNCLASSIFIED if a value is not provided
Type feeType
Attributes minOccurs=0

period - (element)

Description The unit of time being one of: • DAY (A single calendar day. No consideration of holidays or weekends) • WEEKDAY (Weekends are taken into consideration but holidays falling on week days are not). CALENDARDAY (A single calendar day. No consideration of holidays or weekends) • BUSINESSDAY (Weekends and holidays are taken into consideration) • WEEK (A calendar week) • MONTH (A calendar month) • MONTHEND (The end of the month from n months away, 1 MONTHEND, is the end of this month) • QUARTER (Three calendar months) • QUARTEREND (The end of the quarter from n quarters away, 1 QUARTEREND, is the end of this quarter, note these are calendar quarters, so March, June, September, December) • YEAR (A calendar year) • YEAREND (The end of the year from n years away, 1 YEAREND is the end of this year)

Type period

Attributes

periodMultiplier - (element)

Description The number of units for this Interval

Type xs:int

Attributes

agreement - (element)

Description The agreement ID, for example an ISDA agreement ID.

Type xs:string

Attributes minOccurs=0

agreementType - (element)

Description The type of agreement, for example an ISDA.

Type xs:string

Attributes minOccurs=0

calendar - (element)

Description The Calendar to use to determine if the 'Spot' or Effective date of an OTC Product is a business day

Type ref

Attributes minOccurs=0

effectiveDate - (element)

Description

Type kasmDate

Attributes minOccurs=0

fees - (element)

Description The list of Fee payments attached to the OTC Product. None, one or many fee payments are supported for an OTC Product
Type
Attributes minOccurs=0

fee - (element)

Description Details of a fee payment for an OTC Product
Type fee
Attributes minOccurs=0
maxOccurs=unbounded

issuer - (element)

Description
Type xs:string
Attributes minOccurs=0

paySwapStream - (element)

Description The swap stream where coupons are being paid from the perspective of the Buyer. The type of InterestRateStream depends on the type of swap. A FixedRateStream represents a swap where the pay leg is based on a fixed rate. A FloatingRateStream represents a swap where the pay leg is based on a floating rate
Type kasmType
Attributes

productSubType - (element)

Description The sub-type of the product, if a further level of definition is needed on the type of a product in addition to the productType field. This can be used for more concise ProductConfig mappings for example when adjustments to the base product types logic or calculation is required.
Type xs:string
Attributes minOccurs=0

productType - (element)

Description The type of Product. This is set automatically by the underlying product class. It should be overridden for specific product types (indicated by Set manually in the Validation column. The supported productType values are shown in the Validation column
Type xs:string
Attributes

receiveSwapStream - (element)

Description The swap stream where coupons are being received from the perspective of the Buyer. The type of InterestRateStream depends on the type of swap. A FixedRateStream represents a swap where the receive leg is based on a fixed rate. A FloatingRateStream represents a swap where the receive leg is based on a floating rate

Type kasmType

Attributes

settlementCashflows - (element)

Description Structure containing a list of SettlementFlows.

Type settlementCashflows

Attributes minOccurs=0

spotBdc - (element)

Description The BusinessDayConvention used to adjust the 'Spot' or Effective date of the OTC Product.

Type businessDayConvention

Attributes minOccurs=0

terminationDate - (element)

Description Deprecated – this is no longer required. The termination date is derived from the product attributes

Type kasmDate

Attributes minOccurs=0

tradedRate - (element)

Description

Type xs:double

Attributes minOccurs=0

accrualBusinessDayConvention - (element)

Description The BusinessDayConvention used to derive the accrual dates of the InterestRateStream. This BusinessDayConvention is also used to derive the payment dates if paymentBusinessDayConvention is not specified.

Type businessDayConvention

Attributes minOccurs=0

accrualCalendar - (element)

Description The reference to the Calendar that is used to adjust InterestRateStream coupon accrual dates. This Calendar is also used to derive payment dates in the event the paymentCalendar is not specified

Type ref

Attributes minOccurs=0

currency - (element)

Description The currency the InterestRateStream is denominated in
Type ref
Attributes

dateGenerationRule - (element)

Description The Rule used to generate the InterestRateStream accrual and payment dates
Type dateGenerationRule
Attributes minOccurs=0

dayCountFraction - (element)

Description The DayCount convention used to calculate payment amounts
Type dayCountFraction
Attributes minOccurs=0

effectiveDate - (element)

Description The unadjusted effective date of the InterestRateStream. Effective date is the start date of the first interest accrual period
Type kasmDate
Attributes

expiryDate - (element)

Description The unadjusted expiry date of the InterestRateStream (also known as maturity or end date)
Type kasmDate
Attributes

finalPrincipalExchange - (element)

Description A true/false flag to indicate whether there is a final exchange of principal on the expiry date. This typically applies to cross currency swaps. • Set to True if the Principal amount is exchanged on the expiry date • Set to False if the Principal amount is NOT exchanged on the expiry date
Type xs:boolean
Attributes minOccurs=0

firstPaymentDate - (element)

Description The firstPaymentDate is used to specify the Unadjusted payment date of the first payment date of the InterestRateStream. This is specified for an InterestRateStream with an irregular first payment. If firstPaymentDate date is specified all subsequent payments are derived from this date rather than from the Effective Date This should be set for an InterestRateStream that has an initial stub payment. For these cases firstPaymentDate should be set to the date of the first InterestRateStream stub payment

Type kasmDate

Attributes minOccurs=0

fixedRateSchedule - (element)

Description The list of Step rate amounts and corresponding dates for this FixedRateStream

Type

Attributes minOccurs=0

step - (element)

Description The fixed rate schedule expressed as outstanding fixed rates and dates.

Type step

Attributes minOccurs=0
maxOccurs=unbounded

frequency - (element)

Description The payment frequency of the InterestRateStream - e.g. Annual, 6-monthly etc

Type interval

Attributes

fullFirstCoupon - (element)

Description The FullFirstCoupon flag indicates if the first InterestRateStream payment should be treated as a full coupon when calculating the interest amount of an initial stub payment Set to true if the interest for the firstCoupon is calculated using the number of days between the firstCoupon accrual end date and an accrual start date derived by applying the frequency to the accrual end date Set to false if interest is calculated using the number of days between the effectiveDate and the firstCoupon accrual end date. Default value is false

Type xs:boolean

Attributes

initialPrincipalExchange - (element)

Description A true/false flag to indicate whether there is an initial exchange of principal on the effective date. This typically applies to cross currency swaps. • Set to True if the Principal amount is exchanged on the effective date • Set to False if the Principal amount is NOT exchanged on the effective date

Type xs:boolean

Attributes minOccurs=0

longFinalStub - (element)

Description Set for an InterestRateStream with a final stub period. Set to true if the penultimate coupon is removed and the final coupon's accrual start date is set to the accrual start date of the removed penultimate coupon Set to false if the final coupon is a short stub. Default value is false

Type xs:boolean

Attributes

notional - (element)

Description The notional amount of the InterestRateStream

Type xs:double

Attributes

notionalStepSchedule - (element)

Description The list of Step Notional amounts and corresponding dates for this InterestRateStream

Type

Attributes minOccurs=0

step - (element)

Description The notional amount schedule expressed as outstanding notional amounts and dates.

Type step

Attributes minOccurs=0
maxOccurs=unbounded

payReceive - (element)

Description Specifies whether the amounts in the InterestRateStream are being paid or received from the perspective of the buyer. This value is set automatically by the system. No value should be supplied when sending a document. If a value is supplied the system will replace it with the correct value. • PAY (Specifies the InterestRateStream amounts are being paid from the perspective of the Buyer) • RECEIVE (Specifies the InterestRateStream amounts are being received from the perspective of the Buyer)

Type payReceive

Attributes minOccurs=0

paymentBusinessDayConvention - (element)

Description The BusinessDayConvention used to derive the Payment dates of the InterestRateStream. This should be set if the Payment date business day convention is different to accrualBusinessDayConvention

Type businessDayConvention

Attributes minOccurs=0

paymentCalendar - (element)

Description The Reference to the Calendar that is used to adjust InterestRateStream payment dates. If this is not explicitly set then the accrualCalendar is used to derive InterestRateStream Accrual AND Payment dates
Type ref
Attributes minOccurs=0

rate - (element)

Description The rate used to calculate each coupon generated from the InterestRateStream parameters. The rate is specified as an absolute amount. e.g. A fixedRate of 0.015 is equivalent to a rate of 1.5%
Type xs:double
Attributes

streamType - (element)

Description The type of InterestRateStream. This is automatically set based on the type of InterestRateStream • FIXED (Interest payments are based on a fixed rate so are known at the start of the interest rate stream) • FLOAT (Interest payments are based on a floating interest rate so are unknown at the start of the interest rate stream)
Type streamType
Attributes minOccurs=0

termStructureConfig - (element)

Description The term structure that is used to discount the InterestRateStream. Typically this is not specified and is derived from configuration data based upon the currency. It will override any configuration data termStructureConfig settings if specified.
Type ref
Attributes minOccurs=0

accrualBusinessDayConvention - (element)

Description The BusinessDayConvention used to derive the accrual dates of the InterestRateStream. This BusinessDayConvention is also used to derive the payment dates if paymentBusinessDayConvention is not specified
Type businessDayConvention
Attributes minOccurs=0

accrualCalendar - (element)

Description The reference to the Calendar that is used to adjust InterestRateStream coupon accrual dates. This Calendar is also used to derive payment dates in the event the paymentCalendar is not specified
Type ref
Attributes minOccurs=0

compoundingMethod - (element)

Description The compounding method used to calculate the Floating Rate interest amounts. Note: This is only used when the refix rate period is more frequent than the InterestRateStream frequency • COMPOUNDING (This Compounding method applies any spread to the floating rate prior to compounding). • FLAT (Flat compounding treats the Floating Rate and Spread differently in different periods. In the current period the interest is calculated using Floating Rate plus spread but in subsequent periods the accumulated interest is calculated using the Floating Rate only). • SIMPLE (The simple method applies any spread to the floating rate after it has been compounded). • NONE (no compounding of rates is applied).

Type compoundingMethod

Attributes minOccurs=0

currency - (element)

Description The Currency the InterestRateStream is denominated in

Type ref

Attributes

dateGenerationRule - (element)

Description The Rule used to generate the InterestRateStream accrual and payment dates. By Default Forward is used

Type dateGenerationRule

Attributes minOccurs=0

dayCountFraction - (element)

Description The DayCount convention used to calculate payment amounts

Type dayCountFraction

Attributes minOccurs=0

effectiveDate - (element)

Description The unadjusted effective date of the InterestRateStream. Effective date is the start date of the first interest accrual period

Type kasmDate

Attributes

expiryDate - (element)

Description The unadjusted expiry date of the InterestRateStream (also known as maturity or end date)

Type kasmDate

Attributes

finalPrincipalExchange - (element)

Description A true/false flag to indicate whether there is a final exchange of principal on the expiry date. This typically applies to cross currency swaps. • Set to True if the Principal amount is exchanged on the expiry date • Set to False if the Principal amount is NOT exchanged on the expiry date

Type xs:boolean

Attributes minOccurs=0

firstPaymentDate - (element)

Description The firstPaymentDate is used to specify the Unadjusted payment date of the first payment date of the InterestRateStream. This is specified for an InterestRateStream with an irregular first payment If firstPaymentDate date is specified all subsequent payments are derived from this date rather than from the Effective Date This should be set for an InterestRateStream that has an initial stub payment. For these cases firstPaymentDate should be set to the date of the first InterestRateStream stub payment

Type kasmDate

Attributes minOccurs=0

floatingTermStructure - (element)

Description A reference to the term structure that is used to derive the floating rate for the Floating Rate stream. Typically this is not specified and is derived from configuration data based upon the currency and refix rate. It will override any configuration data termStructureConfig settings if specified.

Type ref

Attributes minOccurs=0

frequency - (element)

Description The payment frequency of the InterestRateStream - e.g. Annual, 6-monthly etc

Type interval

Attributes

fullFirstCoupon - (element)

Description The FullFirstCoupon flag indicates if the first InterestRateStream payment should be treated as a full coupon when calculating the interest amount of an initial stub payment Set to true if the interest for the firstCoupon is calculated using the number of days between the firstCoupon accrual end date and an accrual start date derived by applying the frequency to the accrual end date Set to false if interest is calculated using the number of days between the effectiveDate and the firstCoupon accrual end date. Default value is false

Type xs:boolean

Attributes

initialPrincipalExchange - (element)

Description A true/false flag to indicate whether there is a initial exchange of principal on the effective date. This typically applies to cross currency swaps. • Set to True if the Principal amount is exchanged on the effective date • Set to False if the Principal amount is NOT exchanged on the effective date

Type xs:boolean

Attributes minOccurs=0

initialRefixDate - (element)

Description The initialRefixDate for the floatingRateStream. The initialRefixDate will have a value if the rule for deriving the initialRefixDate is different to the rule for deriving all subsequent refix dates

Type kasmDate

Attributes minOccurs=0

initialRefixDateOffset - (element)

Description The Interval that is applied to first coupon's accrual start date to determine the refix date for the first floating coupon This is set if the refix interval for the first coupon is different to the interval defined for the subsequent refix rates

Type interval

Attributes minOccurs=0

initialRefixRate - (element)

Description The initial refix rate to be used as an override to the observed refix rate. This refix rate just applies to the first floating coupon and if specified the first floating coupon will be derived from this rate rather than the observed refix rate

Type xs:double

Attributes minOccurs=0

longFinalStub - (element)

Description Set for an InterestRateStream with a final stub period Set to true if the penultimate coupon is removed and the final coupon's accrual start date is set to the accrual start date of the removed penultimate coupon Set to false if the final coupon is a short stub. Default value is false

Type xs:boolean

Attributes

notional - (element)

Description The notional amount of the InterestRateStream

Type xs:double

Attributes

notionalStepSchedule - (element)

Description The list of Step Notional amounts and corresponding dates for this InterestRateStream

Type

Attributes minOccurs=0

step - (element)

Description The notional amount schedule expressed as outstanding notional amounts and dates.

Type step

Attributes minOccurs=0
maxOccurs=unbounded

payReceive - (element)

Description Specifies whether the amounts in the InterestRateStream are being paid or received from the perspective of the buyer. This value is set automatically by the system. No value should be supplied when sending a document. If a value is supplied the system will replace it with the correct value. • PAY specifies the InterestRateStream amounts are being paid from the perspective of the Buyer • RECEIVE specifies the InterestRateStream amounts are being received from the perspective of the Buyer

Type payReceive

Attributes minOccurs=0

paymentBusinessDayConvention - (element)

Description The BusinessDayConvention used to derive the Payment dates of the InterestRateStream. This should be set if the Payment date business day convention is different to accrualBusinessDayConvention

Type businessDayConvention

Attributes minOccurs=0

paymentCalendar - (element)

Description The Reference to the Calendar that is used to adjust InterestRateStream payment dates. If this is not explicitly set then the accrualCalendar is used to derive InterestRateStream Accrual AND Payment dates

Type ref

Attributes minOccurs=0

refixCalendar - (element)

Description Reference to the Calendar that is used to adjust floatingRateStream refix dates

Type ref

Attributes minOccurs=0

refixDateOffset - (element)

Description . This is used to calculate fixingDate. If a swap refixes differently to 2 business days prior to accrual start date then this field would be used to specify this.
Type interval
Attributes minOccurs=0

interestRateDef - (element)

Description Reference to the InterestRateDef that defines the interest rate to be used to derive the refix rate for each coupon of the floatingRateStream
Type ref
Attributes

spread - (element)

Description The spread that is applied to the floatingRateStream refix rate The spread is specified in Basis Points. e.g. a Spread of 15 equates to 0.15%. Can be positive or negative
Type xs:double
Attributes minOccurs=0

spreadSchedule - (element)

Description The list of Step spread amounts and corresponding dates for this InterestRateStream
Type
Attributes minOccurs=0

step - (element)

Description The spread rates and dates.
Type step
Attributes minOccurs=0
maxOccurs=unbounded

streamType - (element)

Description The type of InterestRateStream. This is automatically set based on the type of InterestRateStream • FIXED (Interest payments are based on a fixed rate so are known at the start of the interest rate stream) • FLOAT (Interest payments are based on a floating interest rate so are unknown at the start of the interest rate stream)
Type streamType
Attributes minOccurs=0

termStructureConfig - (element)

Description The term structure that is used to discount the InterestRateStream.
Type ref
Attributes minOccurs=0

agreement - (element)

Description The agreement ID, for example an ISDA agreement ID.
Type xs:string
Attributes minOccurs=0

agreementType - (element)

Description The type of agreement, for example an ISDA.
Type xs:string
Attributes minOccurs=0

calendar - (element)

Description The Calendar to use to determine if the 'Spot' or Effective date of an OTC Product is a business day
Type ref
Attributes minOccurs=0

currency - (element)

Description The currency the Repo is denominated in
Type ref
Attributes minOccurs=0

dayCountFraction - (element)

Description The DayCount convention for the Repo interest rate. The DayCount convention specifies how the interest rate should be treated when calculating interest amounts
Type dayCountFraction
Attributes minOccurs=0

description - (element)

Description A free text field that is passed-through to the BondSettlementInstruction
Type xs:string
Attributes minOccurs=0

effectiveDate - (element)

Description The unadjusted Effective (also known as Purchase) Date of the Repo. This is the date the first leg of the Repo is settled
Type kasmDate
Attributes

expiryBusinessDayConvention - (element)

Description The Business day convention used to adjust the Repo product expiry date
Type businessDayConvention
Attributes minOccurs=0

expiryCalendar - (element)

Description
Type ref
Attributes minOccurs=0

expiryDate - (element)

Description The Expiry (also known as Repurchase) Date of the Repo product. The Expiry date is the unadjusted end date of the Repo agreement On this date the lent securities are returned and the product expires
Type kasmDate
Attributes

fees - (element)

Description The list of Fee payments attached to the OTC Product. None, one or many fee payments are supported for an OTC Product
Type
Attributes minOccurs=0

fee - (element)

Description Details of a fee payment for an OTC Product
Type fee
Attributes minOccurs=0
maxOccurs=unbounded

notional - (element)

Description The notional amount of the Repo
Type xs:double
Attributes

productType - (element)

Description The type of Product. This is set automatically by the underlying product class. It should be overridden for specific product types (indicated by Set manually in the Validation column. The supported productType values are shown in the Validation column
Type xs:string
Attributes

productSubType - (element)

Description The sub-type of the product, if a further level of definition is needed on the type of a product in addition to the productType field. This can be used for more concise ProductConfig mappings for example when adjustments to the base product types logic or calculation is required.
Type xs:string
Attributes minOccurs=0

issuer - (element)

Description
Type xs:string
Attributes minOccurs=0

purchaseProceeds - (element)

Description The purchase proceeds for the Security - i.e. the cash amount exchanged for the Purchase leg of the Repo
Type xmlAmount
Attributes

repoType - (element)

Description The type of the Repo (default is REPO). The type specifies who is the lender and borrower of the securities. For REPO trades, by convention, the Seller of the Repo agreement sells the securities on the effective date of the Repo and buys the securities back on the expiry date of the Repo) For REVERSREPO trades, by convention, the Seller of the Repo agreement buys the securities on the effective date of the Repo and sells the securities back on the expiry date of the Repo)
Type repoType
Attributes minOccurs=0

repurchaseProceeds - (element)

Description The repurchase proceeds for the Security - i.e. the cash amount exchanged for the Repurchase leg of the Repo
Type xmlAmount
Attributes

security - (element)

Description The Security which is sold and repurchased as part of this Repo agreement
Type ref
Attributes

settlementCashflows - (element)

Description Structure containing a list of SettlementFlows.
Type settlementCashflows
Attributes minOccurs=0

spotBdc - (element)

Description The BusinessDayConvention used to adjust the 'Spot' or Effective date of the OTC Product.
Type businessDayConvention
Attributes minOccurs=0

terminationDate - (element)

Description Deprecated – this is no longer required. The termination date is derived from the product attributes
Type kasmDate
Attributes minOccurs=0

termStructureConfig - (element)

Description The reference to the term structure that is used to value the Repo. The specified term structure is used to discount the future Repo payment amounts to determine its current Net Present Value (NPV). Typically this is not specified and is derived from configuration data based upon the currency. It will override any configuration data termStructureConfig settings if specified.
Type ref
Attributes minOccurs=0

volume - (element)

Description The volume of securities traded.
Type xs:int
Attributes minOccurs=0

currency - (element)

Description
Type xs:string
Attributes minOccurs=0

originalAmounts - (element)

Description
Type
Attributes minOccurs=0

amount - (element)

Description
Type xmlAmount
Attributes minOccurs=0
maxOccurs=unbounded

style - (element)

Description
Type amountStyle
Attributes minOccurs=0

value - (element)
Description
Type xs:string
Attributes minOccurs=0

settlementCashflow - (element)
Description The abstract SettlementFlow object.
Type settlementFlow
Attributes maxOccurs=unbounded

actualSettlementDate - (element)
Description The date on which the trade actually settles.
Type kasmDate
Attributes minOccurs=0

calendar - (element)
Description
Type ref
Attributes

expectedSettlementDate - (element)
Description The date on which the trade is expected to settle.
Type kasmDate
Attributes

stepDate - (element)
Description The unadjusted date from when the stepValue is effective. On each stepDate the associated stepValue becomes effective.
Type kasmDate
Attributes

stepValue - (element)
Description The fixed rate value that becomes effective from the corresponding stepDate until the next stepDate is reached. This is specified as a double amount rather than a percentage. e.g. A fixed rate of 0.015 is equivalent to a rate of 1.5%. Can be positive or negative.
Type xs:double
Attributes

agreement - (element)
Description
Type xs:string
Attributes minOccurs=0

agreementType - (element)

Description
Type xs:string
Attributes minOccurs=0

calendar - (element)

Description
Type ref
Attributes minOccurs=0

currencyRef - (element)

Description
Type ref
Attributes minOccurs=0

effectiveDate - (element)

Description The unadjusted Effective (also known as Purchase) Date of the Bond. This is the date the Bond is due to settle and the proceeds are paid
Type kasmDate
Attributes

fees - (element)

Description
Type
Attributes minOccurs=0

fee - (element)

Description
Type fee
Attributes minOccurs=0
maxOccurs=unbounded

notional - (element)

Description The notional amount of the Bond
Type xs:double
Attributes

productType - (element)

Description
Type xs:string
Attributes

productSubType - (element)
Description
Type xs:string
Attributes minOccurs=0

issuer - (element)
Description
Type xs:string
Attributes minOccurs=0

purchaseProceeds - (element)
Description The purchase proceeds for the Security - i.e. the cash amount exchanged for the Bond securities
Type xmlAmount
Attributes

security - (element)
Description The Security which is sold or purchased with this bond trade
Type ref
Attributes

settlementCashflows - (element)
Description
Type settlementCashflows
Attributes minOccurs=0

spotBdc - (element)
Description
Type businessDayConvention
Attributes minOccurs=0

terminationDate - (element)
Description
Type kasmDate
Attributes minOccurs=0

termStructureConfig - (element)

Description The reference to the term structure that is used to value the Bond. The specified term structure is used to discount the future Bond payment amount to determine its current Net Present Value (NPV). Typically this is not specified and is derived from configuration data based upon the currency. It will override any configuration data termStructureConfig settings if specified.
Type ref
Attributes minOccurs=0

volume - (element)

Description The volume of securities traded.
Type xs:int
Attributes minOccurs=0

Code4Text - Qualifier, code (simple type)

Description Qualifier, code
Type Derivative of type: xs:string
Format Min.=4
Max.=4

FunctionOfMessage - Function of message (simple type)

Description Function of message
Type Derivative of type: xs:string
Format

ISODate - Date (simple type)

Description Date
Type Derivative of type: xs:date
Format

ISODateTime - Date and time (simple type)

Description Date and time
Type Derivative of type: xs:dateTime
Format

KDPWMemberIdentifier - KDPW member identifier (simple type)

Description KDPW member identifier
Type Derivative of type: xs:string
Format Min.=4
Max.=4

Max16Text - Maximum 16 characters text (simple type)

Description Maximum 16 characters text
Type Derivative of type: xs:string
Format Min.=1
Max.=16

Max35Text - Maximum 35 characters text (simple type)

Description Maximum 35 characters text
Type Derivative of type: xs:string
Format Min.=1
Max.=35

NumberLong - Number in long data type (simple type)

Description Number in long data type
Type Derivative of type: xs:long
Format

refType - (simple type)

Description
Type Derivative of type: xs:string
Format

buySell - (simple type)

Description
Type Derivative of type: xs:string
Format

kasmDateTime - (simple type)

Description
Type Derivative of type: xs:string
Format

kasmDate - (simple type)

Description
Type Derivative of type: xs:string
Format

dayCountFraction - (simple type)

Description
Type Derivative of type: xs:string
Format

businessDayConvention - (simple type)

Description
Type Derivative of type: xs:string
Format

period - (simple type)

Description
Type Derivative of type: xs:string
Format

dateGenerationRule - (simple type)

Description
Type Derivative of type: xs:string
Format

payReceive - (simple type)

Description
Type Derivative of type: xs:string
Format

streamType - (simple type)

Description
Type Derivative of type: xs:string
Format

compoundingMethod - (simple type)

Description
Type Derivative of type: xs:string
Format

amountStyle - (simple type)

Description
Type Derivative of type: xs:string
Format

repoType - (simple type)

Description
Type Derivative of type: xs:string
Format

feeType - (simple type)

Description
Type Derivative of type: xs:string
Format

otcd.ntf.001.01 - OTC notification (complex type)

Description OTC notification
Component elements GnlInf
MsgData

KDPWDocument - KDPW system message (complex type)

Description KDPW system message
Component elements otcd.ntf.001.01

GeneralInformation - General information (complex type)

Description General information
Component elements SndrMsgRef
FuncOfMsg
CreDtTm
SeqNb
NtfTp

DateAndDateTimeChoice - Date and time (complex type)

Description Date and time
Component elements Dt
DtTm

MsgData - OTC market processing notification (complex type)

Description OTC market processing notification
Component elements contents

contents - (complex type)

Description
Component elements content

automaticTerminationSummary - Message broadcast to the Clearing Members informing them of changes to their portfolio of Trades as a result of the Automatic Termination process. (complex type)

Description Message broadcast to the Clearing Members informing them of changes to their portfolio of Trades as a result of the Automatic Termination process.

Component elements
- canceledTrades
- participant
- externalAccountId
- internalAccountId
- updatedTrades

kasmType - (complex type)

Description

Component elements

tradeDetail - Used by the AutomaticTerminationSummary and ParticipantNotification messages. (complex type)

Description Used by the AutomaticTerminationSummary and ParticipantNotification messages.

Component elements
- participant
- clientId
- counterpartyId
- counterpartyClientId
- defaultAccountApplied
- externalDealId
- externalId
- internalDealId
- positionAccountExternalId
- positionAccountInternalId
- reason
- source
- status
- trade

trade - Represents the trade and underlying product details of a financial transaction. (complex type)

Description Represents the trade and underlying product details of a financial transaction.

Component elements
account
analysisTag
buySell
deal
externalTradeDescriptor
externalTradeIdentifier
features
lastUpdated
originalTradeId
product
tradeDate
tradeState
trader

kasmEntity - (complex type)

Description

Component elements

externalTradeDescriptor - (complex type)

Description

Component elements

ref - (complex type)

Description

Component elements
cacheNameOverride
revisionNumber

tag - (complex type)

Description

Component elements

transientRef - (complex type)

Description

Component elements

fra - A Fra instance defines the economic details of a Fra (Forward Rate Agreement). The Buyer receives the Floating rate and pays the Fixed rate. The Seller pays the Floating rate and receives the Fixed rate. (complex type)

Description A Fra instance defines the economic details of a Fra (Forward Rate Agreement). The Buyer receives the Floating rate and pays the Fixed rate. The Seller pays the Floating rate and receives the Fixed rate.

Component elements
agreement
agreementType
calendar
currency
dayCountFraction
effectiveBusinessDayConvention
effectiveCalendar
effectiveDate
expiryBusinessDayConvention
expiryCalendar
expiryDate
fees
floatingTermStructure
fraTenor
issuer
notional
productSubType
productType
refixDate
interestRateDef
settlementCashflows
spotBdc
termStructureConfig
terminationDate
tradedRate

fee - (complex type)

Description

Component elements
amount
businessDayConvention
calendar
currency
payReceive
paymentDate
type

interval - (complex type)

Description

Component elements
period
periodMultiplier

swap - The Swap class specifies the economic and other product parameters that are specific to a Swap trade. A Swap is made up of multiple InterestRateStreams. The combination of InterestRateStream types (FixedRateStream or FloatingRateStream) and their parameters dictates the type of Swap e.g. Interest Rate, OIS, Basis, Cross Currency etc. The Buyer of a Swap pays the coupons of the paySwapStream and receives the coupons of the receiveSwapStream. The Seller of a Swap receives the coupons of the paySwapStream and pays the coupons of the receiveSwapStream. (complex type)

Description The Swap class specifies the economic and other product parameters that are specific to a Swap trade. A Swap is made up of multiple InterestRateStreams. The combination of InterestRateStream types (FixedRateStream or FloatingRateStream) and their parameters dictates the type of Swap e.g. Interest Rate, OIS, Basis, Cross Currency etc. The Buyer of a Swap pays the coupons of the paySwapStream and receives the coupons of the receiveSwapStream. The Seller of a Swap receives the coupons of the paySwapStream and pays the coupons of the receiveSwapStream.

Component elements

agreement
agreementType
calendar
effectiveDate
fees
issuer
paySwapStream
productSubType
productType
receiveSwapStream
settlementCashflows
spotBdc
terminationDate
tradedRate

fixedRateStream - A FixedRateStream is used to specify the fixed leg of a Swap. The appropriate paySwapStream or receiveSwapStream should be set as a FixedRateStream if the Swap has a fixed payment stream. FixedRateStream is a subclass of InterestRateStream. (complex type)

Description A FixedRateStream is used to specify the fixed leg of a Swap. The appropriate paySwapStream or receiveSwapStream should be set as a FixedRateStream if the Swap has a fixed payment stream. FixedRateStream is a subclass of InterestRateStream.

Component elements

accrualBusinessDayConvention
accrualCalendar
currency
dateGenerationRule
dayCountFraction
effectiveDate
expiryDate
finalPrincipalExchange
firstPaymentDate
fixedRateSchedule
frequency
fullFirstCoupon
initialPrincipalExchange
longFinalStub
notional
notionalStepSchedule
payReceive
paymentBusinessDayConvention
paymentCalendar
rate
streamType
termStructureConfig

floatingRateStream - A FloatingRateStream is used to specify the floating leg of a Swap. The appropriate paySwapStream or receiveSwapStream should be set as a FloatingRateStream if the Swap has a floating payment stream. For Basis swaps both the paySwapStream and receiveSwapStream will be set as a FloatingRateStream. FloatingRateStream is a subclass of InterestRateStream. (complex type)

Description A FloatingRateStream is used to specify the floating leg of a Swap. The appropriate paySwapStream or receiveSwapStream should be set as a FloatingRateStream if the Swap has a floating payment stream. For Basis swaps both the paySwapStream and receiveSwapStream will be set as a FloatingRateStream. FloatingRateStream is a subclass of InterestRateStream.

Component elements

accrualBusinessDayConvention
accrualCalendar
compoundingMethod
currency
dateGenerationRule
dayCountFraction
effectiveDate
expiryDate
finalPrincipalExchange
firstPaymentDate
floatingTermStructure
frequency
fullFirstCoupon
initialPrincipalExchange
initialRefixDate
initialRefixDateOffset
initialRefixRate
longFinalStub
notional
notionalStepSchedule
payReceive
paymentBusinessDayConvention
paymentCalendar
refixCalendar
refixDateOffset
interestRateDef
spread
spreadSchedule
streamType
termStructureConfig

repo - A Repo instance defines the economic details of a Repo (Repurchase Agreement). The Buyer of a RepoType REPO buys the securities on the effective date and sells the securities back on the expiry date. The Seller of a RepoType REPO sells the securities on the effective date and buys the securities back on the expiry date. (complex type)

Description A Repo instance defines the economic details of a Repo (Repurchase Agreement). The Buyer of a RepoType REPO buys the securities on the effective date and sells the securities back on the expiry date. The Seller of a RepoType REPO sells the securities on the effective date and buys the securities back on the expiry date.

Component elements

agreement
agreementType
calendar
currency
dayCountFraction
description
effectiveDate
expiryBusinessDayConvention
expiryCalendar
expiryDate
fees
notional
productType
productSubType
issuer
purchaseProceeds
repoType
repurchaseProceeds
security
settlementCashflows
spotBdc
terminationDate
termStructureConfig
volume

xmlAmount - (complex type)

Description

Component elements

currency
originalAmounts
style
value

settlementCashflows - SettlementCashflows contains a list of “SettlementFlow” objects, which contain the details of expected and actual settlement dates, as well as the settlement amount or Security and Security Volume. (complex type)

Description SettlementCashflows contains a list of “SettlementFlow” objects, which contain the details of expected and actual settlement dates, as well as the settlement amount or Security and Security Volume.

Component elements
- settlementCashflow

settlementFlow - The abstract SettlementFlow object. (complex type)

Description The abstract SettlementFlow object.

Component elements
- actualSettlementDate
- calendar
- expectedSettlementDate

step - (complex type)

Description

Component elements
- stepDate
- stepValue

bond - A Bond instance defines the economic details of a Bond or Bond Forward trade. The Buyer of a Bond buys the securities on the effective date. The Seller of a Bond sells the securities on the effective date. (complex type)

Description A Bond instance defines the economic details of a Bond or Bond Forward trade. The Buyer of a Bond buys the securities on the effective date. The Seller of a Bond sells the securities on the effective date.

Component elements
- agreement
- agreementType
- calendar
- currencyRef
- effectiveDate
- fees
- notional
- productType
- productSubType
- issuer
- purchaseProceeds
- security
- settlementCashflows
- spotBdc
- terminationDate
- termStructureConfig
- volume