



FIX REFERENCE DATA for AIX

Version: 1.0

Publication Date: 26.09.2025



Genium INETSM

All content in this document is owned, or licensed, by Nasdaq, Inc. or its affiliates ('Nasdaq'). Unauthorized use is prohibited without written permission of Nasdaq. While reasonable efforts have been made to ensure that the contents of this document are accurate, the document is provided strictly "as is", and no warranties of accuracy are given concerning the contents of the information contained in this document, including any warranty that the document will be kept up to date. Nasdaq reserves the right to change details in this document without notice. To the extent permitted by law no liability (including liability to any person by reason of negligence) will be accepted by Nasdaq or its employees for any direct or indirect loss or damage caused by omissions from or inaccuracies in this document.

Copyright © 2018 Nasdaq, Inc. All rights reserved.

CONTENTS

References	6
Overview	7
Supported messages	7
Administrative messages	7
Inbound Application messages.....	7
Outbound Application messages	7
Streaming Reference Data Over Fix.....	8
AIX FIX Repository	8
The FIX Session	9
CompIDs	9
SenderSubID	9
User Authentication.....	9
Renewal of passwords	9
Expired passwords.....	10
Logon	10
Heartbeat intervals.....	10
Encryption	10
Datatypes and required fields	10
Character encoding.....	10
Session lifetime	11
Failover and message recovery	11
The Standard Header	11
The Standard Trailer.....	12
Message Details.....	12
How to interpret the Required (Req'd) column.....	12
Logon – inbound to Marketplace	12
Logon – outbound from Marketplace.....	13
Logout (in/out)	14
Sequence Reset (in/out).....	14
Resend Request (in/out)	14
Reject (out).....	15
Heartbeat (in/out)	15
Test Request (in/out)	15
APPLICATION SEQUENCING	17
Application Sequencing Details	17

Application IDs	17
The ApplicationSequenceControl group	17
The ApplIDRequestGrp.....	18
Requesting And Recovering Reference Data	18
Limitations to Reference Data recovery	18
Application Message Request	18
Application Message Request Ack.....	19
Reference Data	21
Introduction	21
Requesting Reference Data	21
Main Workflow	21
Security Definition	21
Market Definition	21
Trading Session List.....	22
Security Status	22
Price Reference.....	22
MESSAGE DETAILS.....	23
Security Definition (out).....	23
Security Definition Request (in).....	27
Security Definition Update Report (out)	28
Market Definition (out)	34
Trading Session List (out)	35
Security Status Request (in)	36
Security Status (out)	36
Price Reference Request (in).....	37
Price Reference (out).....	37
Application Message Request (in).....	38
Application Message Request Ack (out).....	39

REVISION HISTORY

Date	Revision	Change Description
26-09-2025	1.0	Initial version.

References

[1] FIX 5.0 SP2 Protocol Specification

<http://fixprotocol.org/specifications/fix5.0sp2spec>

Overview

This document contains the specification for the FIX interface to the AIX trading system. The interface is based on the FIX Protocol 5.0 SP2 standard (Financial Information exchange). More detailed information about the standard can be found in FIX specification document see [1].

The interface implemented by AIX follows the FIX specifications as far as possible. In the absolute majority of cases the structure and semantics of the messages are identical to the standard. In some cases, the protocol has been extended to cover functions not considered by the standard.

These extensions are clearly detailed in the document. In other cases, the standard is ambiguous or indicates that the details should be bilaterally agreed between the parties. In these cases this manual provides a detailed description to avoid any possible ambiguity.

Supported messages

Administrative messages

Logon

Logout

Sequence Reset

Resend Request

Reject

Heartbeat

Test Request

Inbound Application messages

Application Message Request

Security Definition Request

Security Status Request

Price Reference Request

Outbound Application messages

Application Message Request Ack

Security Definition

Security Definition Update Report

Market Definition

Trading Session List

Security Status

Price Reference

Streaming Reference Data Over Fix

AIX FIX Reference Data over FIX is implemented using Application Sequencing.

AIX FIX Repository

This specification defines the full set of messages, fields and enumerated values that can be used. As with most FIX implementations, this only supports a small subset of all available messages, components, fields and enumerated values defined in FIX 5.0 SP2. An FPL-formatted repository corresponding to this specification is delivered separately.

NOTE: Inbound messages not conforming to this spec, will be rejected with a session-level Reject message.

The FIX Session

The session layer offers limited standard FIX session support. Ordinary FIX message recovery is not supported. The Resend Request message is supported, but it will always be responded to with a Sequence Reset – gap fill message.

Message recovery is instead supported via application sequencing. See chapter 4 for details.

CompIDs

The Sender- and TargetCompID uniquely define the FIX session. A session can only be active (established) between two hosts simultaneously. Any attempts to establish a second FIX session using the same CompIDs (for instance to a backup gateway) in parallel will be rejected.

- The TargetCompID for transactions sent inbound to the Exchange will be “AI” for production and “AI_TEST” for test systems.
- The SenderCompID for transactions sent outbound from the Exchange will be “AI” for production and “AI_TEST” for test systems.

SenderSubID

Each inbound business transaction must have the SenderSubID (tag 50) field set to an authenticated user. One user can be authenticated by setting the Username and Password field in the Logon message. Additional users can be authenticated using the User Request message. See chapter 4 for a description on how to authenticate additional users.

The SenderSubID on incoming transactions will be echoed back in TargetSubID (tag 57) on outbound transactions.

NOTE: On the Logon or User Request, the SenderSubID must be set to the user id the client intends to log on.

User Authentication

Each incoming business transaction must have a username set in the SenderSubID field. The user needs must be authenticated for the transaction to be accepted. A user is authenticated by setting the Username (553) and Password (554) fields in the Logon message.

Renewal of passwords

A new password may be set by setting the NewPassword (925) field along with the current password in the Password (554) field. This can be done either with the Logon message or the User Request message. The SessionStatus (1409) field of the Logon returned to the client can be checked to see if the new password was accepted.

Expired passwords

If the password has expired when a client tries to log in, the system will respond with a Logout message with SessionStatus set to 8 – Password expired. To gain access, the client must issue a new Logon message with NewPassword set (along with the expired password in Password). If the new password is not valid, the system will respond with another Logout message. SessionStatus will be set to 3 – New session password does not comply with policy. The client will be able to log in again with another new password.

Logon

At Logon, clients are identified by:

- CompIDs
- IP Address

The Logon Username and Password fields are used to authenticate the client. When the client is authenticated, the system responds with a Logon message to the client.

Heartbeat intervals

Heartbeat intervals are negotiated at Logon using the HeartBtInt (108) field. The system allows heartbeat intervals greater than 10 seconds. Recommended heartbeat interval is 30 s. A heartbeat interval set lower than 10 seconds will result in a Logout response.

Encryption

The system does not support encryption.

Datatypes and required fields

This specification does not change the datatype on any fields defined in the standard FIX specification. There may be places where this specification restricts the value range of a field further than specified in standard FIX. This will be clearly marked in the spec.

All fields listed in this specification that are marked as required in the standard specification, are required also in this specification. This document specifies additional fields as required. These fields are marked with a ‘Q’ in the required column of the message listings.

Character encoding

The FIX gateway will use the 8-bit standard ISO-8859-9 encoding, often called Latin-9. The lower 7 bits are compatible with the standard 7-bit ASCII character encoding.

Session lifetime

The FIX session lifetime is restricted to one trading day. Unlike ordinary FIX sessions the sequence number restarts at 1 after a disconnect or Logout. Message recovery using standard FIX Resend Requests is not supported. Application Sequencing is used for message recovery.

Failover and message recovery

Message recovery in this solution is based on Application Sequencing. A client must implement Application Sequencing in order to perform message recovery. For backward compatibility reasons the ordinary FIX session-level message recovery transactions are still supported. But the response to a Resend Request will be an empty Sequence Reset –gap fill message. See chapter 4 for details on Application Sequencing.

All FIX sessions have at least one primary and one secondary gateway to which the session states are fully replicated. This means that regardless to which gateway a client connects, full message recovery is provided.

A client cannot have the same FIX session active towards multiple gateway instances simultaneously.

Failover is as simple as establishing a connection to a backup gateway, and perform message recovery as described above.

NOTE: A client is not allowed to have simultaneous active connections to both a primary and a backup gateway.

The Standard Header

All FIX messages contain a Standard Header. The header contains important information such as session identifiers (CompIDs), sequence numbers and message type and length etc.

Tag num	FIX Field name	Req'd	Comment
8	BeginString	Y	Identifies beginning of new message and protocol version. ALWAYS FIRST FIELD IN MESSAGE. Valid values: FIXT.1.1
9	BodyLength	Y	Message length, in bytes, forward to the CheckSum field. ALWAYS SECOND FIELD IN MESSAGE.
35	MsgType	Y	Defines message type ALWAYS THIRD FIELD IN MESSAGE.
49	SenderCompID	Y	As specified in separate agreement
50	SenderSubID		Required on inbound transactions. Must be set to a valid authenticated user.
56	TargetCompID	Y	As specified in separate agreement
57	TargetSubID		Should not be populated on inbound transactions. Will contain the value of incoming SenderSubID on outbound transactions. In some cases, such as in unsolicited cancels, TargetSubID will not be set.
34	MsgSeqNum	Y	Integer message sequence number.

Tag num	FIX Field name	Req'd	Comment
43	PossDupFlag		Indicates possible retransmission of message with this sequence number. Always required for retransmitted messages
97	PossResend		Indicates that message may contain information that has been sent under another sequence number. Required when message may be duplicate of another message sent under a different sequence number.
52	SendingTime	Y	Time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT"))
122	OrigSendingTime		Original time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT")). Required for message resent as a result of a ResendRequest.

The Standard Trailer

All FIX messages end with a Standard Trailer. The trailer only includes a simple checksum field. The details on how to calculate the checksum can be found in the standard FIX specification.

Tag num	FIX Field name	Req'd	Comment
10	Checksum	Y	

Message Details

How to interpret the Required (Req'd) column

A 'Y' marks the field as required in standard FIX (and of course also in this implementation). A 'Q' means that the field is required in this implementation although it is not required in standard FIX. No entry at all means the field is optional.

Logon – inbound to Marketplace

The response to a logon is either a Logon, which denotes a successful logon, or a Logout. A client must be prepared to handle failure scenarios including (but not limited to):

A Logon attempt may fail or be rejected for several reasons. The FIX gateway will react differently depending on the kind of failure. The two different actions it may take are:

- Silently ignore the Logon.
If authentication fails (for security reasons).
If the wrong Sender or Target CompID is specified.
For other reasons specified in the standard FIX specifications.
If the FIX gateway has no connection with the back-end system.
- Respond with a Logout.
Logon failure for other reasons than authentication/security.
The Logout response to a Logon will always contain a note on why in the Text (58) field.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. NOTE: Resetting the sequence numbers will result in all prior messaging being lost. Valid values: Y = Yes
108	HeartBtInt	Y	Heartbeat interval. Any value greater than 10 s is accepted. A lower value will result in a Logout response.
553	Username	Q	User name
554	Password	Q	NOTE: Must be in CAPITAL LETTERS. password (unencrypted)
925	NewPassword		Specifies a new password for the FIX Logon. The new password is used for subsequent logons.
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2
	Standard Trailer	Y	

Logon – outbound from Marketplace

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. Will only be set as a response to an inbound Logon with this flag set. Valid values: Y = Yes
108	HeartBtInt	Y	As specified in inbound Logon. Valid range: Greater than 10 s
1409	SessionStatus	Q	Status of the FIX session. Valid values: 0 = Session Active 1 = Session password changed 3 = New session password does not comply with policy 9= Invalid Sendersubid or sendercompid 100 = Invalid BodyLength, session suspended 101 = HeartBt interval too low
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2
20002	DaysToPwdExpiry		Number of days to password expiration.

Tag num	FIX Field name	Req'd	Comment
	Standard Trailer	Y	

Logout (in/out)

The Logout message is used to gracefully disconnect a FIX session. When receiving a Logout, the counterparty should respond with a Logout. A Logout can also be the response to an unsuccessful Logon attempt.

SessionStatus = 100 means that a critical formatting error has been detected in an inbound transaction. The gateway is unable to reliably continue parsing further messages on the session. The connection is closed and can only be enabled by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 5
1409	SessionStatus		Status of the FIX session. Only set on outbound Logouts. Valid values: 3 = New session password does not comply with policy 4 = Session logout complete 5 = Invalid password or user name 6 = Account locked 8 = Password expired 9 = Invalid Sendersubid or sendercompid 100 = Invalid BodyLength, session suspended 101 = HeartBt interval too low
58	Text		Free text
	Standard Trailer	Y	

Sequence Reset (in/out)

This message has two uses. The common usage is with GapFillFlag set to 'Y', which is used in a response to a Resend Request to indicate that a range of messages will not be resent. This is commonly used to avoid resending administrative messages like Heartbeats.

The other (very rare) usage is to reset the sequence numbers to a higher number to get out of a deadlock. This is only triggered by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 4
123	GapFillFlag		
36	NewSeqNo	Y	
	Standard Trailer	Y	

Resend Request (in/out)

Resend Request is not supported for FIX Reference Data.

Reject (out)

The Reject, or session-level reject, message is sent whenever the FIX gateway is able to at least partially parse the message, but the message does not adhere to the specification and cannot be delivered to the back-end system.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 3
45	RefSeqNum	Y	
371	RefTagID		
372	RefMsgType		
373	SessionRejectReason	Q	Valid values: 0 = Invalid Tag Number 1 = Required Tag Missing 2 = Tag Not Defined For This Message Type 3 = Undefined Tag 4 = Tag Specified Without A Value 5= Value Is Incorrect Out Of Range For This Tag 6 = Incorrect Data Format For Value 9 = CompID Problem 10 = SendingTime Accuracy Problem 11 = Invalid MsgType 15 = Repeating group fields out of order 16 = Incorrect NumInGroup count for repeating group 99 = Other
58	Text		
	Standard Trailer	Y	

Heartbeat (in/out)

A heartbeat message is sent at the interval set at Logon. It is also the response to a Test Request message.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 0
112	TestReqID		Identifier included in Test Request message to be returned in resulting Heartbeat. Required when the heartbeat is the result of a Test Request message.
	Standard Trailer	Y	

Test Request (in/out)

Test Request is used to “ping” the counterparty whenever a heartbeat has not arrived at the negotiated heartbeat interval.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 1
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat
	Standard Trailer	Y	

APPLICATION SEQUENCING

Application Sequencing Details

FIX Application Sequencing is a new concept introduced in FIX 5.0. It allows for a more fine-grained subscription and recovery, where the receiver can dictate what content will be sent. As such, it is very suitable for Reference Data dissemination.

The enabling of reference data is initiated by the client, which sends an Application Message Request.

Application IDs

The Reference Data is separated into logical streams, called an Application. The Application is assigned a unique Application ID. Each Application is sequence numbered separately. When logged in, the client requests enabling/recovery of reference data by sequence number for each Application. Note that all Applications are sent over the same FIX session.

The Application ID used for Reference Data is:

APPL ID	TYPE OF DATA	COMMENT
R	Reference Data	A single Application ID for all reference data.

The ApplicationSequenceControl group

Every Reference Data message contains the ApplicationSequenceControl group. The group contains the Application ID and sequence number used in recovery. The receiver needs to track the sequence number for each application. ApplID (1180) contains the Application ID. Tag 1181, ApplSeqNum contains the sequence number.

ApplLastSeqNum contains the sequence number of the last sequence number sent for this Application ID on the current session. This allows gaps in the sequence. A receiver must check this field to avoid unnecessary resends.

The ApplResendFlag (1352) is only set on the responses to a request for resending of Application IDs that support full recovery. Full recovery is not available for Reference Data (see section 4.2.1 for details).

TAG NO	TAG NAME	COMMENT
1180	ApplicationSequenceControl/ ApplID	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Sequence number for this Application ID.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Last sequence number for this Application ID. Used to indicate gaps in the sequence.

1352	ApplicationSequenceControl/ ApplResendFlag	Set to Y if this message is a result of a resend. Valid values: Y = Yes N = No (default if not present)
------	---	--

The ApplIDRequestGrp

The ApplIDRequestGrp is a repeating group in the Application Message Request message that contains the requested sequence numbers for each Application ID. The following fields are used:

TAG NO	TAG NAME	COMMENT
1351	ApplIDRequestGrp/NoApplIDs	Number of Application IDs in this request.
1355	ApplIDRequestGrp/RefApplID	Application ID. Valid values: R = Reference Data
1182	ApplIDRequestGrp/ApplBegSeqNum	First requested sequence number. Ignored for non-recoverable Application IDs.
1183	ApplIDRequestGrp/ApplEndSeqNum	Only 0 (zero) is allowed (subscription will always be enabled).

Requesting And Recovering Reference Data

In this solution Reference Data is sent to the client after an Application Message Request has been issued.

Limitations to Reference Data recovery

The Application Message Request also contains the application sequence numbers which governs recovery of lost data. Note that for reference data, full recovery is not supported. A request will result in enabling the real-time transmission of the data for the Application IDs in the request. The real-time data is preceded with an initial snapshot providing the current state. Regardless of the sequence numbers given in the request, the response will always start at the next sequence number with the real-time data (preceded with a possible snapshot).

Application Message Request

An Application Message Request is a general request to enable reference data as well as to recover lost reference data.

The message contains a repeating group with one entry for each Application ID (see section 4.1.1 for a list of available applications). Each Application ID present in the message enables the “subscription” on that type of data (given that the user is authorized to see said data).

Request Sequence Numbers

For each Application ID it is also possible to supply a start and end application sequence number to recover lost messages.

ApplEndSeqNum (1183) is used to define the end of a set of messages to recover. If it is set to 0 it means that a subscription is enabled. In this solution a request for an Application ID always enables the subscription, ApplEndSeqNum must be set to 0.

NOTE: As a consequence of always enabling a subscription, subsequent requests for the same Application ID after Logon will result in a reject.

ApplBegSeqNum (1182) is used to set the beginning of the messages to be sent.

- For the non-recoverable Application IDs this value is ignored. A snapshot will always be sent prior to the real-time messages.
- For the fully recoverable Application IDs, this value will dictate the starting point of the recovery. If ApplBegSeqNum is higher than the last sequence number, real-time messages will be enabled without any recovery.

NOTE: The receiver must keep track of the sequence numbers on each Application ID received to be able to recover in any situation where messages have been lost.

NOTE 2: Complete message recovery for all reference data is not possible.

The Response to an Application Message Request is an Application Message Request Ack. If the request was successful, the Ack will be followed by Reference Data messages.

Application Message Request Ack

The Application Message Request Ack (request ack) message is the response to an Application Message Request.

The ApplResponseType field (1348) signals if the request was successful or not.

Successful Requests

For a completely successful request, the request ack will contain:

- ApplResponseType = 0 (Request successfully processed)

Following the Application Message Request Ack, the messages will always be sent in the following order:

1. Market Definitions
2. Trading Session List
3. Security Definitions

4. Security Statuses

5. Price References

NOTE: A request may fail for some Application IDs, but still be successful for other.

Requests Failing For One Or More Application Ids

If a request is made for a non-existent application id, the request ack will contain:

- ApplResponseType = 1 (application does not exist)
- ApplResponseError = 0 (application does not exist) for that Application ID.

If a request is made for an application already requested previously, the request ack will contain:

- ApplResponseType = 3 (Duplicate requests for application)
- ApplResponseError = 3 (Duplicate requests for application) for that Application ID.

If a request is made for an application with ApplEndSeqNum (1183) not set to 0 (zero), the following will be returned in the request ack:

- ApplResponseError = 1 for that Application ID.

Reference Data

Introduction

The systems allow the transmission of Security Definitions, Market Definitions, Security Statuses, Price References and Trading Session Lists. To enable Reference Data the receiver logs in and sends an Application Message Request.

An Application Message Request is a general request for reference data. The same message is also used to recover lost messages. A successful Application Message Request returns one or more Reference Data messages.

The System also supports requests for Security Status, Security Definition and Price Reference which will send out information for a single instrument without setting up a subscription. The unsolicited indicator (tag 325) will indicate if the Security Definition, Security Status or Price Reference message was sent out as part of a subscription or not.

Requesting Reference Data

In this solution Reference Data is sent to the client after an Application Message Request has been issued. All Reference Data messages are enabled by this single request. See chapter 4 for details.

The System also supports requests for Security Statuses, Security Definitions and Price References without setting up a subscription by sending in a Application Message Request.

Main Workflow

Security Definition

The Security Definition is used to publish start-of-day reference data for each tradable security in the system. For intra-day updates the Security Definition Update Report is used. Price field and corporate action updates will not trigger a Security Definition Update Message.

Market Definition

The Market Definition message is used to publish information on the market structure of the marketplace. Each tradable security belongs to one market (represented by one Market Definition message).

Trading Session List

The Trading Session List message contains all trading states (Trading Sessions) the instruments can be in. It contains information on Trading Rules, Matching Rules, and allowed order types for each state.

TradingSessionID (336) contains the actual ID of the state. Security Status messages also contain this ID to identify the state it refers to. This is in contrast to standard FIX where TradingSessionID contains enums such as DAY, HALFDAY etc.

Security Status

All session State change information is sent on the instrument level using the Security Status message. The Security Status message also contains information about last traded price and corporate actions.

Price Reference

Price reference contains reference price, current upper and lower limits, base price and previous day closing price for an instrument.

For price limits both the static and current price limits will be considered when this message is sent.

The disseminated range will be the smallest allowed range.

Example:

The static price limits are: 10.00 – 10.50

The dynamic price limits are: 10.10 – 10.60

Result: The disseminated price limits are: 10.10 – 10.50

When there is fixed matching there is no allowed price limit range. In such a case the upper and lower limit values sent in fix will be the same as the base price. When there is no price limits the upper and lower price limit tags will not be sent at all.

MESSAGE DETAILS

Security Definition (out)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = d
1180	ApplicationSequenceControl/ ApplID	Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
55	Symbol	Q	Short name.
107	SecurityDesc		Long name
48	SecurityID	Q	Orderbook ID
22	SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
167	SecurityType		Valid values: 1 = Options 2 = Forward 3 = Futures 4 = FRA 5 = Cash 6 = Payment 7 = Exchange Rate 8 = Interest rate swap 9 = REPO 11 = Combination 12 = Guarantee 13 = OTC general 14 = Equity warrant 15 = Security lending 17 = Strip 18 = Certificates
541	MaturityDate		Date of Maturity.
20041	NoCouponBlock		Number of listed coupons
→	223	CouponRate	
→	224	CouponPaymentDate	
306	UnderlyingIssuer		
231	ContractMultiplier		Specifies the ratio or multiply factor to convert from “nominal” units (e.g. contracts) to total units (e.g. shares).
200	MaturityMonthYear		Specifies the month and year of maturity. Format: YYYYMM

TAG	FIX TAG NAME			REQ'D	COMMENT
202	StrikePrice				Strike Price
15	Currency				Currency
20035	LastTradingDate				Date of Last Trading
20037	FirstTradingDate				Date of First Trading
20036	LastTradingTime				Time of Last Trading
20038	FirstTradingTime				Time of First Trading
20039	SectorCode				Sector Code
20040	SectorCodeDesc				Sector Code Description
225	IssueDate				
873	DatedDate				
454	NoSecurityAltID				Number of alternate Security Identifiers
	455	SecurityAltID			International Securities Identification Number (ISIN)
	456	SecurityAltIDSource			Valid values: 4 = ISIN number
1310	NoMarketSegments			Q	Number of market segments on which the security is traded. Will always be 1.
→	1301	MarketID		Q	Name of the market
→	1300	MarketSegmentID			Market Segment Code
→	1396	MarketSegmentDesc			Market Segment Name
→	1205	NoTickRules			Number of Tick Rules NOTE: Tick Rules are associated with the security, not with the market.
→	→	1206	StartTickPrice Range		Starting price range for specified tick increment
→	→	1207	EndTickPrice Range		Ending price range for the specified tick increment
→	→	1208	TickIncrement		Tick increment for stated price range. Specifies the valid price increments at which a security can be quoted and traded
	1234	NoLotTypeRules			Number of Lot Types NOTE: Lot Types are associated with the security, not with the market.
→	→	1093	LotType		Lot Type. Valid values: 2 = Round Lot
→	→	1231	MinLotSize		Lot size of lot type specified in LotType(1093). To enter an order for this particular Lot Type MatchIncrement needs to be set to this value. Note that order quantity must be a multiple of this value.
→	→	21010	LotStartPrice		
→	→	21012	LotMaxTrade Vol		The maximum order quantity that can be submitted for a security. It is valid only for Normal Limit Price Orders.
→	→	21013	LotMinQty		The minimum quantity of an order for the defined price level

TAG	FIX TAG NAME			REQ'D	COMMENT
→	→	21068	LotMinQuote		The minimum quote quantity that can be submitted for a security.
→	→	21069	LotMaxQuote		The maximum quote quantity that can be submitted for a security.
→	→	21009	LotDisplayHighQty		The maximum show quantity allowed in an order for the defined price level
→	→	21014	LotDisplayLowQty		The minimum show quantity allowed in an order for the defined price level
→	→	21011	LotDefaultQtyLimit		Default limit in case no price source series or closing price can be found
228	Factor				Price quotation factor
1150	TradingReferencePrice				Reference price.
21003	BasePrice				Base price.
21004	TradingMethod				The two character information field from the Instrument Class.
21005	GrossSettlement				Y if gross settlement is applied on the Instrument Class level, otherwise N.
916	StartDate				For Debt market series, -Value1 date For Equity market series, -First trading date for extra-ordinary series. None if no date is defined.
917	EndDate				For Debt market series, -Value2 date For Equity market series, Last trading date for extra-ordinary series. None if no date is defined.
325	Unsolicited Indicator				Indicates whether the message is sent as a result of a subscription request or not. Will be 'N' in initial startup request to FIX client. Otherwise 'Y'.
21007	TradingSessionName				Name of the Trading Session
21008	PartitionId			Q	Which partition this security is traded on. Always 1 for AIX.
21026	DerivativeLevel				Specifies the level of derivation for the instrument. Valid values: 0 = Spot
21027	NumberOfSharesIssued				Specifies the total number of shares issued for this security.
21028	PhysicalDelivery				Specifies whether the instrument is defined with physical delivery or not. Valid values: Y/N
21029	Multiplier				This is an optional field that provides additional information for Warrants/Certificates.

TAG	FIX TAG NAME	REQ'D	COMMENT
21030	Reference		This is an optional field that provides additional information for Warrants/Certificates.
21032	InstrumentType	Q	Instrument Type
21031	SeriesDesc	Q	Valid values: S – Standard Instrument
965	SecurityStatus	Q	Valid values: 1 = Active 2 = Inactive
1148	LowLimitPrice		Current lower limit for price limits.
1149	HighLimitPrice		Current upper limit for price limits.
5011	OrderMaxValue		Maximum order value, only applicable for equity market.
38	OrderQty		
159	AccuredInterestAmt		Amount of Accrued Interest
1948	CouponFrequencyPeriod		Time unit multiplier for the frequency of fixed income instruments coupon payment.
1949	CouponFrequencyUnit		Time unit associated with the frequency of fixed income instruments coupon payment.
1950	CouponDayCount		The day count convention used in interest calculations for fixed income instruments. Valid values: 1-US30360 4-EU30360 6- ACT360 7- ACT365 8- ACTAFB 9- ACTACT ISMA 17- ACT364
21059	CouponInterest		The coupon interest for the underlying. (Only applicable for interest-related fixed-income instruments.)
21055	OddFirstCoupon		Y if odd-first coupon is used, otherwise N.
21056	OddLastCoupon		Y if odd-last coupon is used, otherwise N.
21057	PremiumUnit		Order entry price unit Valid values: 1 – Price 2– Yield 3 – Point 7 – Basis point 8 – Clean price 9 – Dirty price 12 – Rate
21058	FixedIncomeType		Type of fixed-income underlying. Valid values: 12 – Type 1 Discounted securities 13 – Type 2 Fixed rate bonds

TAG	FIX TAG NAME		REQ'D	COMMENT
				14 – Type 3A Variable rate bonds next coupon is known 15 – Type 4 Index linked strips 16 – Type 5 Index linked bonds 17 – Type 6 Euro bonds USD 18 – Type 7 Euro bonds EUR 19 – Type 8A Irregular Clean price 20 – Type 3B Variable rate bonds next coupon is unknown 21 – Type 8B Irregular Dirty price
21050	NoTradeReports			Number of Trade Reports.
→	21051	TradeReportCode		Number to be used in tag 828 TrdType in Trade Reports. Please refer to the list of supported Trade Type values supplied by the marketplace.
→	21052	MinTradeRepQty		Minimum quantity in Trade Report.
→	21053	MaxTradeRepQty		Maximum quantity in Trade Report.
→	21064	TrLotsize		Trade Report Lotsize
60	TransactTime			Dissemination time = Current time
21063	DaysToMaturity / Repo Term			Number of days to maturity date
8000	ShortSaleRestriction			Specify what Short Sell validation rule that should be used. Valid values; 0 = No restrictions (Short sell is allowed with no price validation) 1 = Security is not shortable (Short sell is not allowed.) 2 = Security not shortable at or below the best bid (Short sell is allowed with Uptick rule price validation)
21065	IndexClassification			Specify a valid Index Classification
21066	ValueDate			Value date for underlying.
21067	ResetDate			
743	DeliveryDate			Delivery Start Date
21072	NumberOfDecimalInPrice			Number of decimals used in price for the order book.
21073	InstrumentClass			The Instrument class identity.
	Standard Trailer		Y	

Security Definition Request (in)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = c
320	SecurityReqID	Y	Client-generated identifier.
321	SecurityRequestType	Y	Type of Security Definition request.

TAG	FIX TAG NAME	REQ'D	COMMENT
			Valid values: 4 = Symbol
55	Symbol	Q	Short name.
	Standard Trailer	Y	

Security Definition Update Report (out)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = BP
1180	ApplicationSequenceControl/ ApplID	Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
980	SecurityUpdateAction	Q	Valid values: A = Add D = Delete M = Modify
55	Symbol	Q	Short name
107	SecurityDesc		Long name
48	SecurityID	Q	Orderbook ID
22	SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
167	SecurityType		Valid values: 1 = Options 2 = Forward 3 = Futures 4 = FRA 5 = Cash 6 = Payment 7 = Exchange Rate 8 = Interest rate swap 9 = REPO 11 = Combination 12 = Guarantee 13 = OTC general 14 = Equity warrant 15 = Security lending 17 = Strip 18 = Certificates

TAG	FIX TAG NAME			R E Q 'D	COMMENT
541	MaturityDate				Date of Maturity.
20041	NoCouponBlock				Number of listed coupons
	223	CouponRate			
	224	CouponPaymentDate			
306	UnderlyingIssuer				
231	ContractMultiplier				Specifies the ratio or multiply factor to convert from “nominal” units (e.g. contracts) to total units (e.g. shares).
200	MaturityMonthYear				Specifies the month and year of maturity. Format: YYYYMM
201	PutOrCall				Indicates whether an option contract is a put or call. Valid values: 0 = Put 1 = Call
202	StrikePrice				Strike Price
15	Currency				Currency
21001	InstClassification				Instrument classification grouping
20035	LastTradingDate				Date of Last Trading
20036	LastTradingTime				Time of Last Trading
20037	FirstTradingDate				Date of First Trading
20038	FirstTradingTime				Time of First Trading
20039	SectorCode				Sector Code
20040	SectorCodeDesc				Sector Code Description
225	IssueDate				
873	DatedDate				
454	NoSecurityAltID				Number of alternate Security Identifiers
	455	SecurityAltID			International Securities Identification Number (ISIN)
	456	SecurityAltIDSource			Valid Values: 4 = ISIN number
1310	NoMarketSegments			Q	Number of market segments on which the security is traded. Will always be 1.
→	1301	MarketID		Q	Name of the Market
→	1300	MarketSegmentID			Market Segment Code.
→	1396	MarketSegmentDesc			Market Segment Name
→	1205	NoTickRules			Number of Tick Rules NOTE: Tick Rules are associated with the security, not with the market.
→	→	1206	StartTickPriceRange		Starting price range for specified tick increment
→	→	1207	EndTickPriceRange		Ending price range for the specified tick increment
→	→	1208	TickIncrement		Tick increment for stated price range. Specifies the valid price increments at which a security can be quoted and traded

TAG	FIX TAG NAME			R E Q 'D	COMMENT
→	1234	NoLotTypeRules			Number of Lot Types NOTE: Lot Types are associated with the security, not with the market.
→	→	1093	LotType		Lot Type. Valid values: 2 = Round Lot
→	→	1231	MinLotSize		Lot size of lot type specified in LotType(1093). To enter an order for this particular Lot Type MatchIncrement needs to be set to this value. Note that order quantity must be a multiple of this value.
→	→	21010	LotStartPrice		
→	→	21012	LotMaxTradeVol		The maximum order quantity that can be submitted for a security. It is valid only for Normal Limit Price Orders.
→	→	21013	LotMinQty		The minimum quantity of an order for the defined price level
→	→	21068	LotMinQuote		The minimum quote quantity that can be submitted for a security.
→	→	21069	LotMaxQuote		The maximum quote quantity that can be submitted for a security.
→	→	21009	LotDisplayHighQty		The maximum show quantity allowed in an order for the defined price level
→	→	21014	LotDisplayLowQty		The minimum show quantity allowed in an order for the defined price level
→	→	21011	LotDefaultQtyLimit		Default limit in case no price source series or closing price can be found
228	Factor				Price quotation factor
292	CorporateAction				Corporate Actions currently active for this security. Valid values: 01 = R.H. Kullandırılarak Bedelli Sermaye Art 02 = R.H. Kullandırılmadan Bedelli Sermaye Art 03 = Bedelsiz Sermaye Artırımı 04 = Sermaye Azaltımı 05 = Birlesme / Devralma 06 = Temettu 99 = Diger Important Note: When multiple corporate actions occur, tag 292 is sent with a space character between multiple values. (i.e. "292=01 03")
1150	TradingReferencePrice				Reference price.
21003	BasePrice				Base price.
21004	TradingMethod				The two character information field from the Instrument Class.

TAG	FIX TAG NAME	R E Q 'D	COMMENT
21005	GrossSettlement		Y if gross settlement is applied on the Instrument Class level, otherwise N.
916	StartDate		For debt market series, -Value1 date For equity market series, -First trading date for extra-ordinary series. None if no date is defined..
917	EndDate		For debt market series, -Value2 date For equity market series, -First trading date for extra-ordinary series. None if no date is defined.
21007	TradingSessionName		Name of the trading session.
21008	PartitionId		Which partition this security is traded on. Always 1 for AIX
21026	DerivativeLevel		Specifies the level of derivation for the instrument. Valid values: 0 = Spot
21027	NumberOfSharesIssued		Specifies the total number of shares issued for this security.
21028	PhysicalDelivery		Specifies whether the instrument is defined with physical delivery or not. Valid values: Y/N
21029	Multiplier		This is an optional field that provides additional information for Warrants/Certificates.
21030	Reference		This is an optional field that provides additional information for Warrants/Certificates.
21032	InstrumentType	Q	Instrument Type
21031	SeriesDesc	Q	Valid values: S – Standard Instrument
965	SecurityStatus	Q	Valid values: 1 = Active 2 = Inactive
1148	LowLimitPrice		Current lower limit for price limits.
1149	HighLimitPrice		Current upper limit for price limits.
5011	OrderMaxValue		Maximum order value, only applicable for equity market.
38	OrderQty		
159	AccruedInterestAmt		Amount of Accrued Interest

TAG	FIX TAG NAME	R E Q 'D	COMMENT
1948	CouponFrequencyPeriod		Time unit multiplier for the frequency of fixed income instruments coupon payment.
1949	CouponFrequencyUnit		Time unit associated with the frequency of fixed income instruments coupon payment.
1950	CouponDayCount		The day count convention used in interest calculations for fixed income instruments. Valid values: 1-US30360 4-EU30360 6- ACT360 7- ACT365 8- ACTAFB 9- ACTACT ISMA 17- ACT364
21059	CouponInterest		The coupon interest for the underlying. (Only applicable for interest-related fixed-income instruments.)
21055	OddFirstCoupon		Y if odd-first coupon is used, otherwise N.
21056	OddLastCoupon		Y if odd-last coupon is used, otherwise N.
21057	PremiumUnit		Order entry price unit Valid values: 1 - Price 2 – Yield 3 - Point 7 - Basis point 8 – Clean price 9 – Dirty price 12 – Rate
21058	FixedIncomeType		Type of fixed-income underlying. Valid values: 12 – Type 1 Discounted securities 13 – Type 2 Fixed rate bonds 14 – Type 3A Variable rate bonds next coupon is known 15 – Type 4 Index linked strips 16 – Type 5 Index linked bonds 17 – Type 6 Euro bonds USD 18 – Type 7 Euro bonds EUR 19 – Type 8A Irregular Clean price 20 – Type 3B Variable rate bonds next coupon is unknown 21 – Type 8B Irregular Dirty price
21050	NoTradeReports		Number of Trade Reports.

TAG	FIX TAG NAME		R E Q 'D	COMMENT
→	21051	TradeReportCode		Number to be used in tag 828 TrdType in Trade Reports. Please refer to the list of supported Trade Type values supplied by the marketplace.
→	21052	MinTradeRepQty		Minimum quantity in Trade Report.
→	21053	MaxTradeRepQty		Maximum quantity in Trade Report.
→	21064	TrLotsize		Trade Report Lotsize
60	TransactTime			Dissemination time = Current time
21063	DaysToMaturity / Repo Term			Number of days to maturity date
8000	ShortSaleRestriction			Specify what Short Sell validation rule that should be used. Valid values; 0 = No restrictions (Short sell is allowed with no price validation) 1 = Security is not shortable (Short sell is not allowed.) 2 = Security not shortable at or below the best bid (Short sell is allowed with Uptick rule price validation)
21065	IndexClassification			Specify a valid Index Classification
21066	ValueDate			Value date for underlying.
21067	ResetDate			
743	DeliveryDate			Delivery Start Date
21072	NumberOfDecimalInPrice			Number of decimals used in price for the order book.
21073	InstrumentClass			The Instrument class identity.
	Standard Trailer		Y	

Market Definition (out)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = BU
1180	ApplicationSequenceControl/ ApplID	Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
1394	MarketReportID	Y	Required in FIX. Will be set, but can be ignored.
1301	MarketID	Y	Market identifier.
	Standard Trailer	Y	

Trading Session List (out)

TAG	FIX TAG NAME			REQ'D	COMMENT
	Standard Header			Y	MsgType = BJ
1180	ApplicationSequenceControl/ ApplID			Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum			Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum			Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
386	NoTradingSessions			Y	Number of Trading Sessions (states) listed in this message
→	336	TradingSessionID		Y	ID of Trading Session
→	1326	TradingSessionDesc		Q	Human-readable name of Trading Session
→	1237	NoOrdTypeRules			Will always be 1
→	→	40	OrdType		Shows whether Market orders are allowed in this state. Valid values: 1 = Market
→	1239	NoTimeInForceRules			Indicates number of allowed distinct order type rules
→	→	59	TimeInForce		Shows whether IOC or FOK orders are allowed in this state. Valid values: 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FOK)
→	1235	NoMatchRules			Will always be 1
→	→	1142	MatchAlgorithm		Required in FIX if group is present. Always set to [N/A]
→	→	574	MatchType		Valid values: 4 = Auto-match
→	21024	OffHoursTrading			Default value. N
	Standard Trailer			Y	

Security Status Request (in)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = e
263	Subscription Request Type	Q	Subscription request type. Valid values: 0 = Snapshot
55	Symbol	Q	Short name.
324	SecurityStatusReqID	Q	Client-generated identifier.
	Standard Trailer	Y	

Security Status (out)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = f
1180	ApplicationSequenceControl/ ApplID	Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
55	Symbol	Q	Short name.
48	SecurityID	Q	Orderbook ID
22	SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
336	TradingSessionID		ID of Session State
326	SecurityTradingStatus		Will be sent out when a security is suspended or resumed. Values: 2 = Trading Halt 3 = Resume
325	Unsolicited Indicator		Indicates whether the message is sent as a result of a subscription request or not. Will be 'N' in initial startup request to FIX client. Otherwise 'Y'.
31	LastPX		Last trade price in the related security in the last session. This price is not updated in real time after every trade.
	Standard Trailer	Y	

Price Reference Request (in)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = pp
55	Symbol	Q	Short name.
	Standard Trailer	Y	

Price Reference (out)

TAG	FIX TAG NAME	REQ'D	COMMENT
	Standard Header	Y	MsgType = pr
1180	ApplicationSequenceControl/ ApplID	Q	Application ID. Valid values: R = Reference Data
1181	ApplicationSequenceControl/ ApplSeqNum	Q	Application sequence number assigned to the message by the application generating the message.
1350	ApplicationSequenceControl/ ApplLastSeqNum	Q	The previous sequence number in the application sequence stream. Permits an application to publish messages with sequence gaps where it cannot be avoided.
55	Symbol	Q	Short name.
48	SecurityID	Q	Orderbook ID
22	SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
325	Unsolicited Indicator	Q	Indicates whether the message is sent as a result of a subscription request or not. Will be 'N' in initial startup request to FIX client. Otherwise 'Y'.
1148	LowLimitPrice		Current lower limit for price limits.
1149	HighLimitPrice		Current upper limit for price limits.
1150	TradingReferencePrice		Reference Price
21003	Base Price		Base Price
21025	TheoreticalPrice		Theoretical Price
140	PrevClosePx		Previous day's closing price
60	TransactTime		Dissemination time = Current time
	Standard Trailer	Y	

Application Message Request (in)

TAG	FIX TAG NAME		REQ'D	COMMENT
	Standard Header		Y	MsgType = BW
1346	ApplReqID		Y	Unique identifier for request
1347	ApplReqType		Y	Type of Application Message Request being made. Valid values: 1 = Subscription to the specified Applications
1351	NoApplIDs			Number of Application IDs in this request
→	1355	RefApplID	Q	Application ID requested. Valid values: R = Reference Data
→	1182	ApplBegSeqNum		Sequence number of first message to be resent.
→	1183	ApplEndSeqNum		Last Sequence number of message to be resent or 0 (zero) for all messages.
	Standard Trailer		Y	

Application Message Request Ack (out)

TAG	FIX TAG NAME		REQ'D	COMMENT
	Standard Header		Y	MsgType = BX
1353	ApplResponseID		Y	Identifier for the Application Message Request Ack
1346	ApplReqID		Q	Identifier of the request associated with this ACK message
1347	ApplReqType		Q	Type of Application Message Request being made. Valid values: 1 = Subscription to the specified Applications
1348	ApplResponseType		Q	Used to indicate the type of acknowledgement being sent. Valid Values: 0 = Request successfully processed 1 = Application does not exist 2 = Messages not available 3 = Duplicate requests for application
1351	NoApplIDs			Number of Application IDs in this request
→	1355	RefApplID		Application ID requested. Valid values: R = Reference Data
→	1182	ApplBegSeqNum		Sequence number of first message to be resent.
→	1183	ApplEndSeqNum		Last Sequence number of message to be resent or 0 (zero) for all messages.
→	1354	ApplResponseError		Valid values: 0 = Application does not exist 1 = Messages requested are not available 3 = Duplicate requests for application
58	Text			
	Standard Trailer		Y	