FIX for AIX

Version: 1.0

Publication Date: 26.09.2025

Genium INET[SM]

CONTENTS

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

[2] FIX Protocol Limited, Market Data Optimization Working Group:

Recommended Practices for Book Management, Version 2.00, 2007

http://fixprotocol.org/documents/2518/MDOWG_Book_Mgt v20.doc

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

User Request

New Order Single

Order Cancel Replace Request

Order Cancel Request

Trade Capture Report

Outbound Application messages

User Response

User Notification

Execution Report

Order Cancel Reject

Business Message Reject

Trade Capture Report

Trade Capture Report Ack

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 conforms to the standard FIX session. Please see the standard FIX specification for additional 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. The session lifetime is not ended at connectivity loss or even Logouts. The sequence numbers are reset to one each morning.

Failover and message recovery

At reconnect and Logon standard FIX message recovery is performed. 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.

Order Suspension/inactivation at connection loss

A FIX session can be configured by the marketplace to automatically suspend all outstanding orders at FIX connection loss. At reconnection the FIX client will be able to cancel the suspended orders.

FIX Session Level Test Cases

This implementation is fully compliant with the session-level test cases specified in the standard FIX

5.0 SP2 Specification, Volume 2, section “FIX Session-level Test Cases and Expected Behaviors”. The only exception is the encryption test cases.

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.
43	PossDupFlag		Indicates possible retransmission of message with this sequence number. Always required for retransmitted messages

Tag num	FIX Field name	Req'd	Comment
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.

Possible Duplicate vs. Possible Resend

The two FIX fields PossDupFlag (43) and PossResend (97) of the Standard Header have different purposes. The PossDupFlag is set on messages retransmitted as a result of a Resend Request. These messages have the original sequence numbers (MsgSeqNum).

PossResend is set on messages resent with a new sequence number. This may be used to resend an order which no response has been received. The gateway will check whether the client identifier (such as the ClOrdID, TradeReportID etc) in the message has been received before. If the client identifier has been seen before, the message will be dropped.

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.

Repeating groups

The fields in a FIX Repeating group are marked in the message listings with an arrow. Example (Parties block):

453	NoPartyIDs		Optional repeating group only used for on behalf of transactions and account model values.

→	448	PartyID		Q	Party identifier.
→	447	PartyIDSource		Q	Valid values:
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified.
→	802	NoPartySubIDs			Number of PartySubIDs present. Only used for PartyRole=Executing Firm. Will always be 1.
→	→	523	PartySubID	Q	Sub-identifier of party. Here Exchange code of the party.
→	→	803	PartySubIDType	Q	Type of PartySubID (523) value

In the above example nested repeating groups can also be seen.

Also notice that the req'd flag on the NumInGroup field (NoPartyIDs, NoPartySubIDs). If it is present (either Y or Q), it means that the whole repeating group will always be present. A Q or Y set on an individual field in a repeating group means that it will always be present if the repeating group is present.

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 **NOTE:** Must be in CAPITAL LETTERS.
554	Password	Q	password (unencrypted)

Tag num	FIX Field name	Req'd	Comment
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.
	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 used to recover messages when a sequence number gap has been detected.

Once a gap is detected and a resend request initiated, server ignores all requests/messages that are sent from the client during the retransmission period.

Tag num	FIX Field name	Req'd	Comment
	Standard	Y	MsgType = 2
	Header		
7	BeginSeqNo	Y	
16	EndSeqNo	Y	
	Standard Trailer	Y	

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	

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.

If required, an authorized user (master user) can disable (suspend) any individual user within same participant/member (firm) and force logout its active session. In addition, a master user can resume (activate) already suspended user within participant.

NOTE: The exchange can assign one or more FIX users of a firm as a master user.

NOTE 2: Status of suspended/resumed user's orders are subject to related business rules and definitions.

Suspension/Activation can be done by "User Request (MsgType=BE)" and "User Response (MsgType=BF)" FIX Protocol messages.

User Request

The User Request message is used to activate or suspend a user.

User Response

The User Response message is sent as a response to a User Request. Examine the UserStatus (926) field to find out if the request was successful.

Password Management

Renewal of passwords

A new password may be set by setting the NewPassword (925) field along with the current password in Password in the User Request message. The UserStatus (926) field of the User Response 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 User Response message with UserStatus set to 101 – Password expired. To gain access, the client must issue a new User Request message with NewPassword set (along with the expired password in Password).

If the new password is not valid, the system will respond with another User Response with UserStatus set to 102 – New session password does not comply with policy. The client will be able to log in again with another new password.

Users across multiple sessions

The back-end does not allow multiple parallel logins for the same user. Whenever an already logged in user attempts to log in a second time, the first is logged out. This is true across protocols as well. If a user X is logged in on an OMNet session, and the same user tries to log in over FIX, the OMNet user session will be logged out. So care must be taken not to try to log in the same user across multiple sessions.

Message Details

User Request (in)

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = BE
923	UserRequestID	Y	Unique identifier for a User Request.
924	UserRequestType	Y	Indicates the action required by a User Request Message. Valid values: 5 = Suspend user 6 = Resume user
553	Username	Y	A valid backend username. **NOTE:** Must be in CAPITAL LETTERS.
554	Password		Not supported
925	NewPassword		Not supported
	Standard Trailer	Y	

User Response (out)

The User Response message is a response to the User Request message.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = BF
923	UserRequestID	Y	Unique identifier for a User Request.
553	Username	Y	A valid backend username.
924	UserRequestType	Y	Indicates the action required by a User Request Message. Valid values: 5 = Suspend user 6 = Resume user
926	UserStatus	Q	Indicates the status of a user. Valid values: 0 = Reject 7 = Suspended 8 = Resumed

927	UserStatusText		A text description associated with a user status.
	Standard Trailer	Y	

User Notification (out)

The User Notification message is an unsolicited user status message.

Tag num	FIX Field name		Req'd	Comment
	Standard Header		Y	MsgType = CB
809	NoUsernames		Q	Number of user names in this message
→	553	Username	Q	A valid backend username.
926	UserStatus		Q	Indicates the status of a user. Valid values: 7= Forced user logout by Exchange
	Standard Trailer		Y	

Business Level Party Identifiers

Overview

All inbound business messages are subject to marketplace authorization and must therefore specify the party being responsible for the business content of the message. Whenever applicable, the party entering the transaction (if different than business responsible) must also be entered. The SenderCompID and SenderSubID are used to identify the party entering the trade (see implicit parties section below).

The FIX Parties block is used for all other parties.

Parties block

This is a repeating block allowing multiple party identifiers to be set. The following fields must be set for each party:

- PartyID (448) = party identifier
- PartyIDSource (447)
    - D = Proprietary/Custom code
- PartyRole (452) = see below

Party Identifier

The PartyID field can contain different types of identifiers. When it contains a member/participant (firm) identifier, the format is as follows:

The party identifier always consists of the two-character market code followed by the up to 5 charcater firm identifier. Example: A participant with a firm ID of XYZ, would have the party identifier of "AIXYZ".

Available market codes:

- AI

Available Party Roles

The following roles are used:

	Business Role	**PartyRole (452)**	**Comment**
Transaction owner = party legally responsible for consequences of the message	Firm	1 = Executing Firm	Implicit for all transactions other than on-behalf-of or trade reports. Reporting party in trade reports.
	Individual user	12 = Executing Trader	Implicit for all transactions other than on-behalf-of.
Transaction Sender = initiator party of the transaction message	Firm	7 = Entering Firm	Firm who has recorded or reported the execution
	Individual user	36 = Entering Trader	Trader who initiates/submits the transaction (i.e. associated with Entering Firm)
Counterparty in Trade Capture Reports	Firm	17 = Contra Firm	Counterparty in Trade Capture Reports.

Order Management

Overnight orders

Clients who wish to send overnight orders need to make sure that the ClOrdID is unique across the entire lifetime of the order. A simple solution is to include a date in the ClOrdID.

Pass-thru fields

AIX primarily offers three fields as pass-thru fields on incoming transactions. The values of those fields are echoed back to the client in subsequent outgoing transactions. The fields are:

Field	Tag	Length	Comment
Account	1	16	This account field contains either the actual account number or various classifications. All possible values are listed below: Account number (i.e.123456789) HOUSE DO FO FORU FOBY
OrderCapacity	528	2	Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
AllocID	70	15	Free-text pass-thru field. Echoed back.

Instrument Identifiers

For any trading system, the correct identification of securities in a FIX message is of utmost importance. There are several fields within each FIX message, incoming or outgoing, that allow for identification of securities. In this implementation two alternative identifiers can be used:

- Symbol (55) which should contain the OMNet short name (ins_id_s) for the security.
- SecurityID (48) containing the Orderbook ID of the security. This is an alternative numeric identifier that can be used instead of Symbol.

NOTE:

- The Orderbook ID identifier is not provided via OMNet Reference Data.
- The Orderbook ID can be different across trading days for the same security.

Main Workflow

New Order

The order workflow starts with the user submitting a New Order Single message. In response an Execution Report is produced. The Execution Report is a reply directed to the sender of the order and will contain details of the order. If the order is rejected the Execution Report will contain relevant error messages.

Fills

When an order is filled the Execution Report will contain details about the fill. In addition, a Trade Capture Report will be produced. The principal differences between the two are:

Execution Reports are messages directed to the sender of the order and are primarily intended for front-office purposes. It captures order status information as well as fills information (if applicable).

Trade Capture Reports are messages capturing the trade as such and is primarily intended for downstream processing. The Trade Capture Report is used to inform a variety of parties about a trade, e.g.: broker back office; clearing firms; clearing houses; depositories and; regulators. As such downstream processing occurs at various locations and for different purposes, the Trade Capture Report message might look slightly different depending on the receiver.

Trade Capture Report messages are also used for a large number of other purposes, including reporting of privately negotiated trades and relaying trades to parties not directly involved in the trade

– but this is outside the scope of this chapter.

Trade Match ID

The TrdMatchID (880) contains the match id generated by the system. TrdMatchID will hold the 64 bit binary match id encoded as a 16 byte hex string.

NOTE:

TrdMatchID is also set in Trade Capture Report confirmation messages.

Order Modification

Order modification is accomplished through the use of the Order Cancel Replace Request message. Despite its name, it represents a modification of the existing order, not removing the old order and replacing it with a new one. However, an order modification is not a delta change to order instructions; the values set in the Cancel Replace represent the requested new order state. An Execution Report will relay the new state of the order.

- Fields not set in the Cancel Replace are assumed to keep their previous values.
- The required fields must be set regardless if they are changed or not.

Order Attributes allowed to change

Although FIX allows for virtually all of the Order attributes to be changed, there are limitations as to what the back-end AIX system allows. The following attributes are allowed to change:

- OrderQty (38)
- Price (44)
- MaxFloor (111)
- TimeInForce (59) together with ExpireDate (432)
- AllocID (70)

NOTE: Any change to the price of an order, or increasing quantities will result in the order losing its priority in the market.

NOTE 2: Modifying an order to TimeInForce = IOC or FoK is not allowed.

NOTE 3: Modifying the price of an order to a zero is not allowed. If a zero price is desired, the order has to be deleted and a new order with price 0 entered.

NOTE 4: If MaxFloor or TimeInForce are not intended to be changed, do not include them in the Cancel Replace message. They may cause the order to loose priority or the Cancel Replace to be rejected.

Restatements

The Execution Report – Restatement message is used for restating the overnight orders (GTC/GTD) in the morning, In this case, the ExecRestatementReason will be set to 1 = GT renewal / restatement (no corporate action).

Unsolicited modification of orders entered via FIX

Orders entered via FIX can be modified via other protocols like Omnet. It may also be possible for the marketplace to modify existing orders. In such an event an Execution Report – Unsolicited Order Update will be sent out over FIX.

Order Cancellation

If the user wishes to cancel a single previously sent order, the Order Cancel Request message is used.

Execution Reports are issued relaying the status of every canceled order.

In some cases orders may be cancelled in the system without prior request by the user. These will be sent as an Execution Report – Unsolicited Cancel to the client.

The system will generate cancel messages (Execution Report –IOC/Fok Order Cancel) for every IOC and FoK order.

The system will generate cancel messages (Execution Report – Market-to-Limit Order Cancel) for Market-to-Limit orders that could not be immediately matched.

The Order Cancel Request cannot be used for partial cancels.

Cancellation of orders not sent in via FIX

It is possible via FIX to cancel orders originally entered via Omnet or by other means. To cancel such an order, the correct OrderID (Omnet order number), instrument identifier (Symbol or SecurityID) and Side need to be supplied. In this case the OrigClOrdID shall be set to “NONE”.

Unsolicited cancellation of orders entered via FIX

Orders entered via FIX may be cancelled via other protocols like Omnet, or possibly by the marketplace. In such an event an Execution Report – Unsolicited Cancel will be sent out over FIX.

Order suspension/inactivation at connection loss

The back-end can be configured to suspend outstanding orders if a FIX session is disconnected for a configurable interval. Three options are available:

- Do not suspend on disconnect
- Suspend all outstanding orders
- Suspend outstanding orders except for overnight orders (GTC/GTD).

Upon reconnection, Execution Reports will be sent out for all suspended orders. The Execution Reports will have OrdStatus set to 9 – Suspended.

Suspended orders may be cancelled using ordinary Order Cancel Request messages. Suspended order cannot be activated again.

NOTE: The Execution Report –Order Suspended will not contain TargetSubID (57).

NOTE 2: Suspended orders will be cancelled at end-of-day.

Order Features

Order Identification

Client Order ID

Any message related to an order (entry, cancellation, modification) sent by the client, must have a unique identifier in the ClOrdID (11) field. As the standard indicates, the uniqueness of these identifiers must be maintained during the trading session. If orders with duration of more than one trading session are used, the sender needs to cater for uniqueness across those.

Once the message is accepted by the trading engine, the client receives the corresponding confirmation message with the same ClOrdID. In cases where the user immediately after sending an

order wants to modify or cancel it, this can be achieved by referring to the initial order in the OrigClOrdID (41) field of the subsequent message.

Client Order IDs when the Firm uses multiple FIX sessions

Firms using multiple front-end trading applications or multiple FIX sessions should be aware of the following:

- In cases where the exchange offers drop copies of Execution Reports to FIX sessions other than the one that submitted the order, those drop copy Execution Reports will not contain a ClOrdID. The reason for excluding the ClOrdID in those cases is that various FIX sessions or the underlying trading applications might use conflicting ClOrdIDs.
- The above may also apply in cases where exchange business operations perform order management on behalf of the order owner.

Order ID

The OrderID (37) field is the order identifier assigned by the marketplace. This identifier is static and stays with the order even when it is modified.

Users are encouraged to provide the OrderID instead of OrigClOrdID (41) on order updates and cancellations whenever possible, i.e. in all cases except for submitting order actions before the new order ack (Execution Report) is received. The OrderID is the preferred identifier for order modification and cancellation as it is the identifier used internally in the trading engine. Use of other identifiers requires a lookup which increases message latency.

Note that the OrigClOrdID field is required in standard FIX both in Cancel Replace messages and Cancels. If you wish to use the OrderID, it is recommended to set the OrigClOrdID to “NONE” (excluding the quotation marks). The system will ignore OrigClOrdID if OrderID is set in a Cancel or Cancel Replace Request.

As use of the OrderID requires the user to wait for an order acknowledgement from the trading engine, immediate actions require the use of the OrigClOrdID (41) reference field. This field could be necessary to identify the order in communications with the market by other means than FIX.

Execution ID

The ExecID (17) field is not an identifier of trades. It is an identifier assigned to each unique Execution Report message produced by the marketplace, without duplicates during the entire FIX session. The ExecID will be an integer value.

As an exception, in combination orders with trade legs more than 15 the ExecID is not unique.

ExecType

When a fill occurs, the ExecType (150) field will be set to F = Trade.

NOTE: Post-trade corrects or reversals will not be represented on Execution Reports. Please refer to Trade Capture Reports for such functionality.

Order States

Order state changes are divulged in Execution Report messages. Every state change is communicated in an Execution Report.

An order can be in the following intermediate states:

- **New**. This state is applicable when an order is accepted by the trading engine and is not immediately transitioned into any other state:
    - The order is put on the book but not (partially) filled
    - The order is held outside the book waiting for activation, e.g. due to a stop condition or for a session change (as e.g. for a Trigger order).
- **Partially filled.**

  The following are final states, indicating that the order is no longer in the book and no longer available for updates or status requests:

- **Rejected**. The order did not pass validation rules.
- **Canceled**. The order was removed from the system due to a cancellation request, or due to TimeInForce reasons.
- **Filled**. The order is completely filled.
- **Expired**. When a GTD order expires.
- **Suspended**. The order was suspended.

Order Types

Order type is set in the OrdType (40) field. Following order types are supported:

- Market
- Limit

Market orders are always executed at the best possible price. A market order will trade through as many price-levels as needed to be fully filled.

In continuous trading a market order cannot be stored in the book. It has to have a TimeInForce of IOC or FoK.

Market orders may be allowed to enter the book in non-matching states. Once the session changes to a matching state, the order will be executed and/or cancelled.

NOTE: Once the order is converted to a Limit order the OrdType field of subsequent Execution Reports will be set to Limit (including the Order Ack), and the Price field set to the price of the execution.

Order Expiry

An order can specify various conditions for when or how it should expire or be automatically removed from the book.

The evening when a GTD order expires, an Execution Report with OrdStatus (39) set to Cancelled will be sent out for that order.

A GTC order can also expire. Example: A GTC order is suspended. If it isn't deleted or reactivated (not possible via FIX) the same day, an ER with OrdStatus set to Cancelled will be sent out the next day.

NOTE: Only if a GTC order expires because the instrument expires intra-day, or if a GTD order is cancelled by a corporate action, an order expired transaction will be sent out.

Supported TimeInForce (59) values:

Value	Name	Comment
0	Day	
1	Good Till Cancel (GTC)	
3	Immediate Or Cancel (IOC)	
4	Fill or Kill (FoK)	
6	Good Till Date (GTD)	GTD orders must have ExpireDate (432) set.

Quantity Conditions

An order can specify various types of quantity conditions.

Reserve Quantity (a.k.a. "Hidden" or "Iceberg") Orders allow users to hide the full size of their order and thereby potentially limit its influence on prices.

MaxFloor (111): Used to indicate the maximum order quantity shown in the public Market Data. NOTE: MaxFloor = 0, a completely hidden order, is not supported in AIX. Setting MaxFloor to zero will make the full order visible.

NOTE 2: MaxFloor has been changed to behave as expected in standard FIX. The expected behavior is for MaxFloor value to be decreased when the order is partially traded. The previous implementation kept MaxFloor on the original value.

Short Sell Orders

The trading system supports entering of Short Sell Orders –orders to sell an asset that one does not own with the expectation of being able to buy it back later at a lower traded price. Set Side (54) to 5 = Short Sell to enter a short sell order. All subsequent Execution Reports messages for that order (except rejects) will have the short sell flag set.

Missing required fields in Rejects

Due to the way the back-end works, certain fields required in standard FIX 5.0 SP2 for application-level rejects will be missing.

For Order rejects (Execution Report – reject), the following required field will not be present: Side (54)

Also note that on Execution Report –reject messages, the Symbol field (55) will be set to "[N/A]".

Business Message Reject

The Business Message Reject is used to report rejections in situations where other reject messages are not available, e.g. when the inbound message does not reach the trading engine due to trading being closed or authorization not sufficient.

NOTE: The user must be prepared to receive this message as an alternative response to all other business messages.

NOTE 2: If the message count reach pacing limit then all the following messages will be rejected with the Business Message Reject.

How to interpret the message details listings

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.

Message Details

New Order Single –inbound to Marketplace (in)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = D
11	ClOrdID	Y	Unique identifier set by the client.
1	Account	Y	Required pass-thru field set by client and echoed back by marketplace.
528	Client Category	Y	Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
111	MaxFloor		For hidden orders.
55	Instrument/Symbol	Y	Symbol
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime	Y	
38	OrderQty	Y	
40	OrdType	Y	Valid values: 1 = Market 2 = Limit
44	Price		Required for Limit orders
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
	Standard Trailer	Y	

Order Cancel Request (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = F
41	OrigClOrdID		Y	Set to "NONE" if using OrderID instead.
37	OrderID			Recommended to be used instead of OrigClOrdID.
11	ClOrdID		Y	Unique identifier set by the client.
55	Instrument/Symbol			OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
54	Side		Y	Vali d valu es: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime		Y	
38	OrderQtyData/OrderQty		Y	**NOTE:** Required in FIX but ignored by the system. Partial cancels are not supported.
	Standard Trailer		Y	

Order Cancel Replace Request (in)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = G
37	OrderID		Recommended to be used instead of OrigClOrdID.
41	OrigClOrdID	Y	ClOrdID of the order to modify/cancel. Set to "NONE" if using OrderID instead.
11	ClOrdID	Y	Unique identifier set by the client.
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
111	MaxFloor		For hidden orders.
55	Instrument/Symbol		Symbol
54	Side	Y	Required in FIX, but not allowed to change Valid values: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime	Y	
38	OrderQty	Y	
40	OrdType	Y	Required in FIX, but not allowed to change Valid values: 1 = Market 2 = Limit
44	Price		Required for Limit orders
59	TimeInForce		Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
	Standard Trailer	Y	

Order Cancel Reject (out)

Purpose:

- Reject of Order Cancel Request. (CxlRejResponseTo = 1)
- Reject of Order Cancel Replace Request. (CxlRejResponseTo = 2)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = 9
37	OrderID	Y	From Cancel, or if CxlRejReason=1 – Unknown order, OrderID will be set to "NONE".
11	ClOrdID	Y	Unique identifier set by the client.
41	OrigClOrdID	Y	ClOrdID of the order to modify/cancel. Will be set to "NONE" for orders not originally entered via FIX, or if the order could not be found.
39	OrdStatus	Y	Valid values: 0 = New 1 = Partially filled 2 = Filled 4 = Canceled 5 = Replaced 8 = Rejected 9 = Suspended C = Expired
60	TransactTime	Q	
434	CxlRejResponseTo	Y	Valid values: 1 = Order cancel request 2 = Order cancel/replace request
102	CxlRejReason		Valid values: 0 = Too late to cancel 1 = Unknown Order 2 = Broker / Exchange Option 6 = Duplicate ClOrdID (11) received
58	Text		Error description
	Standard Trailer	Y	

Execution Report – Order Ack (out)

Purpose:

- Order Acknowledgement. (ExecType = 0 or L)
- Order reject. (ExecType = 8)
- Cancel of IOC or FOK order. Will always be sent last in a sequence following any immediate fills. If an order is cancelled due to cross trade time out, message shall be sent including"Internal crossing delete" text in tag 58. (ExecType = 4 AND TimeInForce = 3 OR 4)
- Acknowledgement of Order Cancel Replace Request. (ExecType = 5)
- Acknowledgement of Order Cancel Request (ExecType = 4)
- Order was cancelled outside of FIX (via other protocol or by the marketplace). Day, GTD, GTS or GTC Order Canceled. For GTD orders the Cancel transaction will be sent at the end of the day the order expired. For GTS orders the Cancel message is sent at order expiry. Order expiry can occur for GTC orders under certain conditions. See Order Expiry section for details (ExecType = 4 AND ExecRestatementReason = 8 OR ExecRestatementReason = 4)

- Order was updated outside of FIX (via other protocol or by the marketplace). (ExecType = D AND ExecRestatementReason = 8)
- Restatement of overnight (GTC/GTD) orders in the morning. (ExecType = D AND ExecRestatementReason = 1)
- Order Fill. (ExecType = F)
- Order Suspended (likely caused by temporary loss of connectivity). (ExecType = 9)
- If a GTD order is cancelled due to a Corporate Action the Expired Transaction will be sent out. (ExecType = C)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	
11	ClOrdID	Q	Unique identifier set by the client.
17	ExecID	Y	
150	ExecType	Y	Valid values: 0 = New 4 = Canceled 5 = Replaced 8 = Rejected 9 = Suspended L= Activated by the system D = Restated F = Trade C = Expired
39	OrdStatus	Y	Valid values: 0 = New 1 = Partially Filled 2 = Filled 4 = Canceled 5 = Replaced 8 = Rejected 9 = Suspended C = Expired
1	Account		
528	OrderCapacity		
55	Instrument/Symbol		
48	Instrument/SecurityID		Assigned by system. Orderbook ID
22	Instrument/SecurityIDSource		Assigned by system. M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit

Tag	FIX tag name	Req'd	Comment
44	Price		
59	TimeInForce	Q	Valid Values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
151	LeavesQty	Y	Will be equal to OrderQty on Order.
14	CumQty	Y	Will be 0 on Order Ack.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
103	OrdRejReason		Valid values: 0 = Broker / Exchange option
378	ExecRestatementReason	Q	Valid values: 4 = Broker option 8 = Market (Exchange) option
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
111	MaxFloor		For hidden orders. Contains currently shown quantity.
58	Text		Error message
	Standard Trailer	Y	

Business Message Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = j
45	RefSeqNum		MsgSeqNum of rejected message
372	RefMsgType	Y	The MsgType of the FIX message being referenced.
380	BusinessRejectReason	Y	Valid values: 0 = Other 1 = Unknown ID 2 = Unknown Security 3 = Unsupported Message Type 4 = Application not available 5 = Conditionally required field missing
58	Text		Free format text describing the error
	Standard Trailer	Y	

Trade Reporting

Introduction

Trades may, subject to regulations or bilateral agreement, be reported to the marketplace in the following cases:

Trades negotiated between market participants without using execution mechanisms provided by the Marketplace

The marketplace can allow trades to be reported using a set of different mechanisms, the mechanisms currently supported over FIX are:

One-Party Report for Matching

Used when both parties report their trade half. The marketplace matches the reports on security, price, quantity and possibly other conditions.

Two-Party Reports

Used when one of the parties report both sides of a trade by agreement between the parties. Generally allowed only when the marketplace can verify that such an agreement exists between the parties.

Identifiers

Trade Report ID [assigned by participant]

The TradeReportID (571) is similar to the ClOrdID used for orders and executions. A unique Trade Report ID must be set on all reported trades (TCR) inbound to the marketplace. If a client wants to cancel a previous Trade Report, he can use the TradeReportRefID to refer to the original TraderReportID. There is one important exception to the analogy of ClOrdIDs. The marketplace sets its own TradeReportIDs on outbound TCRs (like confirmed trades).

Trade Report Reference ID [assigned by participant]

The TradeReportRefID (572) is used to refer to a previous TCR. A submitter of a reported trade can use TradeReportRefID in subsequent cancellations to the reported trade. The marketplace, which sets its own TradeReportIDs on outbound trade confirmations, uses the TradeReportRefID to reference the submitters TradeReportID from the original trade report, for example on confirmations to reported trades.

Secondary Trade Report ID [assigned by ME]

This ID (818) is set by the marketplace on Trade Capture Report Ack messages. It is an interim identifier assigned to the trade that is valid until the trade is confirmed. The Secondary Trade Report ID carries the Order ID assigned by ME. Analogous to the OrderID on Orders, this is the preferred

identifier to use when canceling a previous Trade Capture Report since it requires no lookup in the gateway. To use it in a Trade Cancel, set SecondaryTradeReportRefID (881) to the value received in 818 in the previous TCR Ack message.

Secondary Trade Report Reference ID [assigned by ME]

The SecondaryTradeReportRefID (881) is the preferred ID to use when canceling a previously reported trade that has not yet been confirmed by the marketplace.

Main Workflow

Trade Capture Report

The Trade Capture Report message is used for the following purposes:

- To submit a new Trade Report (one-party or two-party)
- To cancel a Trade Report
- For the marketplace to publish trade confirmations
- For the marketplace to publish updates to previous trade confirmations
- For the marketplace to notify the contra party when a one-party report has been sent in.

NOTE:

Update Trade Report is not supported.

Submitting a new Trade Report

The TCR message is used to submit off-exchange negotiated trades to the marketplace. Trade Reporting is limited to two models:

- The one-party report for matching model, where both parties report the trade to the marketplace. The marketplace always responds with a Trade Capture Report Ack accepting or rejecting the trade report. When both parties have submitted their side of the trade it is matched by the marketplace and a confirmed trade (also using TCR) is issued.
- The two-party report model, where one party reports for both sides. An agreement must be in place between the parties. The marketplace always responds with a Trade Capture Report Ack accepting or rejecting the trade report. If the report is accepted, a trade confirmation is sent to both parties.

Trade Types

The TrdType tag (828) is used to specify the type of trade being reported to the marketplace.

A list of supported Trade Type values is supplied separately by the marketplace. Note that there may be limitations on which trade types are allowed for a certain instrument and/or participant. It is out of

scope of this document to fully specify all such rules. Please refer to the member trading rules for further information.

Marketplace notification to counterparty

When a one-party report for matching is first entered, the marketplace will send a TCR as a notification to the counterparty.

If the counterparty cancels the trade report causing the notification, a Delete Notification to counterparty will be sent out.

When the trade report causing the notification gets matched, a Delete Notification to counterparty will be sent out.

NOTE: Notification to counterparty messages will not contain the TargetSubID (57).

Marketplace publication of Confirmed Trades

The marketplace uses the TCR to publish confirmed trades, whether auto-matched or reported by clients.

Trade Capture Report Acknowledgement

The TCR Ack is used to acknowledge or reject a Trade Capture Report submitted to the exchange. You will always receive a Trace Capture Report Ack when reporting a trade.

Message Details

Trade Capture Report

Purpose:

- One-Party Report for Matching (in)
- Trade Capture Report – Two-Party Report (in)
- Trade Report cancel (in)

Tag	FIX tag name			Req'd	Comment
	Standard Header			Y	MsgType = AE
571	TradeReportID			Y	Client-generated identifier
487	TradeReportTransType			Y	Valid values: 0 = New 1 = Cancel
856	TradeReportType			Q	Valid values: 0 = Submit 1= Alleged
828	TrdType			Q	For valid values, please see "Appendix Trade Types"
570	PreviouslyReported			Y	Default value. Required in FIX but ignored. N = No
55	Instrument/Symbol				Symbol
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
75	TradeDate			Y	Trade Date. Must be set to a valid date.
60	TransactTime			Y	Time of execution/order creation
552	NoSides			Y	(Repeating Group) Set to 1, only counterparty given Set to 2 for two-party reports
→	54	Side		Y	Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX, but ignored
→	453	NoPartyIDs		Q	
→	→	448	PartyID	Q	PartyID
→	→	447	PartyIDSource	Q	Defualt value : D = Proprietary Code

Tag	FIX tag name			Req'd	Comment
→	→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 17 = Contra Firm
→	1	Account		Y	Account number
→	528	OrderCapacity		Y	Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
→	70	AllocID			Free-text pass-thru field.
	Standard Trailer			Y	

Trade Capture Report Ack (out)

Purpose:

- Ack
- Reject
- Notification to Counterparty (out)
- Trade Report Cancel (out)
- Trade Report Cancel Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
818	SecondaryTradeReportID		Order ID assigned by ME
487	TradeReportTransType		Valid values: 0 = New 1 = Cancel
150	ExecType	Y	Valid values: 0 = Accepted 4 = Canceled 8 = Rejected
939	TradeRptStatus	Q	Valid values: 0 = Accepted 1 = Rejected
856	TradeReportType		Valid values: 0 = Submit 1 = Alleged
55	Instrument/Symbol	Q	Symbol

48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
1	Account		Account number
528	OrderCapacity		Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
70	AllocID		Free-text pass-thru field.
751	TradeReportRejectReason		Valid values: 1 = Invalid party information 2 = Unknown Instrument 3 = Unauthorized To Report Trades 4 = Invalid Trade Type 99 = Other
58	Text		Can contain error message
	Standard Trailer	Y	

Appendix A, Trade types

The following table contains the definitions of all values the TrdType (828) field can contain. (List will be detailed by AIX)

Value	Description
501	LT0 - LAST TRADED PRICE
502	LT6 - LAST TRADED PRICE +/- 6%

Appendix B, Field length limitations

The following fields have a max length limit:

Tag Num	Field Name	max length	Comment
11	ClOrdID	20	
41	OrigClOrdID	20	
70	AllocID	15	
1	Account	15	
528	OrderCapacity	2	
448	PartyID	7	When PartyRole=Executing Firm or Contra Firm
571	TradeReportID	20	
572	TradeReportRefID	20	
881	SecondaryTradeReportRefID	20	
923	UserRequestID	20	
925	NewPassword	32	