



FairX

FIX Drop Copy API Specification

Version 0.07



Table of Contents

Contents

Revision History	4
Overview	5
The FairX Hours of Operation	5
Certification	5
Firm Identifiers	5
Order Identifiers	6
Done for Day and Expiration Reports	6
Self-Match Prevention	6
Session Protocol	7
Messages Format	7
Standard Header Component	7
Standard Trailer Component	8
Administrative Messages	9
Logon (MsgType = A)	9
Logout (MsgType = 5)	9
Resend Request (MsgType = 2)	10
Sequence Reset (MsgType = 4)	10
Test Request (MsgType = 1)	10
Heartbeat (MsgType = 0)	11
Reject (MsgType = 3)	11
Drop Copy Messages	12
Overview	12
Parties Repeating Group	12
Execution Report: Trade (MsgType = 8, ExecType = F)	13



Execution Report: Trade Cancel (MsgType = 8, Tag 39 = H)	15
Execution Report: Accepted Order (MsgType = 8, ExecType = 0 or 4 or 5)	17
Execution Report: Order is Rejected (MsgType=8, ExecType = 8)	19
Execution Report: Done for Day (MsgType = 8, ExecType = 3)	21
Execution Report: Expired (MsgType = 8, ExecType = C)	23
Tag Values	25
Order Status (tag 39)	25
Order Types (tag 40)	25
Order Sides (tag 54 and tag 624)	25
Order Time-in-force (tag 59)	26
Position Effect (tag 77)	26
Cancel Reject Reason (tag 102)	26
Order Reject Reason (tag 103)	26
Execution Types (tag 150)	26
Security Types (tag 167)	27
Session Reject Reason Codes (tag 373)	27
ExecRestatementReason (tag 378)	27
MultiLegReporting (tag 442)	27
OrderCapacity (tag 528)	28
CustOrderCapacity (tag 582)	28
CustomerOrderHandlingInst (tag 1031)	28
AggressorIndicator (tag 1057)	28
Manual or Automated flow (tag 1028)	29
SelfMatchPreventionStrategy (tag 8000)	29



1. Revision History

Version	Date	Author	Description
0.01	Dec-4-2019	ET	Initial draft
0.02	Jun-10-2020	FY	Execution Report: Add Parties Group Added TrdMatchID ExecType updates in the heading
0.03	Jul-9-2020	ET	Updated Tag58
0.04	Dec-7-2020	MG	Removed Memo (5149) from all ExecutionReport messages Updated Fill ExecType=F instead of 1,2 Corrected Execution Type table values
0.05	Dec-8-2020	ET	Updated SelfMatchPreventionStrategy Updated Firm Identifiers Removed Good Till Date (59)
0.06	Mar-4-2021	FY	Update SelfMatchPreventionID to numeric
0.07	Mar-17-2021	VD	Section 2.3 Removing SenderSubID/TargetSubID



2. Overview

This document describes the FairX Drop Copy FIX API for brokers and market makers. The API allows connected firms to receive real-time copies of FairX Execution Report and Acks that are sent over the Order entry API.

The Drop Copy service aggregates sessions/subfirms, allowing customers to aggregate positions and monitor orders across multiple clearing firms.

The API uses FIX protocol version 4.4. This document describes the messages supported by FairX. The document is not intended to serve as a full-fledged specification of the FIX protocol. Please refer to the official FIX specifications at <https://www.fixtrading.org> website for additional details.

2.1. The FairX Hours of Operation

Contact the exchange for the current trading schedule. Orders entered outside trading hours will be rejected. Firms are encouraged to stay connected 15 minutes after the official close to receive execution reports that are generated due to trading session closing logic (e.g. Expired reports, Done for Day).

The exchange will reset its FIX sequences on a weekly basis. Reset schedule is configured during initial setup.

2.2. Certification

In order to connect to FairX, firms must be certified. FairX provides a separate environment for integration, acceptance testing and certification. Please contact the FairX team to obtain additional information.

2.3. Firm Identifiers

All messages sent from/to the exchange *must* contain both the SenderCompID (49) and TargetCompID (56) fields. The firm and the exchange agree on these values at the time of the firm onboarding.

Field	Firm to Exchange	Exchange to Firm
SenderCompID	Id of connection. A firm may have multiple connections, each connection will have its own CompID provided by FAIRX	Always "FAIRX"
TargetCompID	Always "FAIRX"	Id of connection. A firm may have multiple connections, each connection will have its own CompID provided by FAIRX

2.4. Order Identifiers

All orders submitted to FairX must have unique ClOrdID (tag 11). The exchange only enforces the uniqueness of the identifier among working orders (for example, GTC and non-triggered stops). Non-unique ids can cause issues with reporting, clearing, and support.

2.5. Done for Day and Expiration Reports

During initial setup a connection may be configured to distribute Done for Day execution reports for all the GTC/GTD orders that remain open after the trading day is closed. It is also possible to configure the API to distribute the Expired execution reports for the DAY orders that have been expired.



2.6. Self-Match Prevention

Self-match prevention functionality allows market participants to prevent Buy and Sell orders for the same account, firm or group of accounts to match with each other. The functionality is optional and is controlled with a pair of FIX tags in incoming orders:

- **SelfMatchPreventionID** (7928) – orders with the same Self Match Prevention ID for the same executing firm will not match. Maximum length of the ID is 8 digits.
- **SelfMatchPreventionStrategy** (8000) – this value defines the strategy of dealing with matching orders if self-match prevention is triggered. The exchange can either cancel the aggressor order, the resting order or both. The exchange uses the strategy from the *aggressor* order to deal with self-matched orders.



3. Session Protocol

Session protocol assures client identification, sequential request processing, session state control and ability to restore the session after downtime. In the scope of a session, all FIX messages are identified by unique integer sequence numbers and are processed in that order. When an incoming sequence number does not match the expected one, the session must be recovered. If incoming sequence number is less than expected and PossDupFlag is not set to Y, then it is considered a fatal error, and the connection is dropped by the server. If the sequence number of incoming message is greater than the next expected number, then Resend Request is issued for missed messages.

The exchange resets the sequences weekly. Reset schedule is configured during initial setup. Resets may also be initiated by a firm sending a Logon message with ResetSeqNumFlag = Y.

Client and server use the following administrative messages:

- **Logon** – initiates (client) or approves (server) session opening.
- **Logout** – initiates or approves session closing.
- **Resend Request** – requests missed fix messages.
- **Sequence Reset – Gap Fill**, must be used instead of resend of administrative messages.
- **Test Request** – used to control the session state. Requires a reply **Heartbeat** message with properly filled TestReqID (112) field.
- **Heartbeat** – used to control session connection state.
- **Reject** – for administrative message reject.

4. Messages Format

All FIX messages, either administrative or business, require standard header and trailer components.

4.1. Standard Header Component

Tag	Name	FIX Type	Req	Description
8	BeginString	String(7)	Y	FIX Version for session, i.e. 'FIX.4.4'
9	BodyLength	Int(6)	Y	Number of bytes in the message body.
35	MsgType	String(2)	Y	Identifies FIX message type.
34	MsgSeqNum	Int(9)	Y	Sequence number
43	PossDupFlag	Boolean(1)	N	Indicates possible retransmission of message with this sequence number. Must be set to 'Y' for messages sent in response to a Resend Request from FairX

49	SenderCompID	String(7)	Y	<p>To Exchange: Connection identifier assigned by FairX. This tag consists of 2 subcomponents: Session ID and SubFirmID Session ID is the left-most 3 characters. The next 3 characters represent SubFirmID. Example: EBR123 Session ID: EBR SubFirmID: 123</p> <p>From Exchange: Value from order entry tag 56-TargetCompID - "FAIRX"</p>
50	SenderSubID	String(18)	Y	<p>Unique identifier for the end trader submitting orders. Tag 50 IDs are issued by an exchange clearing member firm (clearing member), their contracted vendors or assignees</p>
52	SendingTime	UTCTimestamp(21)	Y	<p>Timestamp of the message leaving the client system.</p>
56	TargetCompID	String(7)	Y	<p>To Exchange: Identifies entity receiving the message = "FAIRX"</p> <p>From Exchange Connection identifier assigned by FairX. This tag consists of 2 subcomponents: Session ID and SubFirmID. Session ID is the left-most 3 characters. The next 3 characters represent SubFirmID. Example: EBR123 Session ID: EBR SubFirmID: 123</p>
57	TargetSubID	String(20)	Y	<p>To Exchange: Identifies destination exchange system - "PROD" or "TEST" From Exchange: Echo back Tag57 sent by client system</p>
122	OrigSendingTime	UTCTimestamp(21)	C	<p>For resent messages only, contains timestamp from tag 52-SendingTime from original message.</p>

4.2. Standard Trailer Component

Tag	Name	FIX Type	Req	Description
10	Checksum	String(3)	Y	<p>Always last tag in message. Functions as end-of-message delimiter.</p>

5. Administrative Messages

5.1. Logon (MsgType = A)

Initiates connection from client side and approves connection if sent by the exchange. ResetSeqNumFlag allows clients to start a new session (reset session sequence numbers).

The Heartbeat Interval is declared by the session initiator using the HeartBtInt field in the Logon message. The heartbeat interval timer should be reset after every message is transmitted (not just heartbeats). The HeartBtInt value should be agreed upon and specified by the Logon initiator (client) and echoed back by the Logon acceptor. The same HeartBtInt value is used by both sides, the Logon “initiator” and Logon “acceptor”.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = “A”
98	EncryptMethod	Int	Y	“0” - None. Security must be guaranteed on transport level
108	HeartBtInt	Int	Y	Heartbeat interval in seconds
141	ResetSeqNumFlag	Boolean	N	“N” - use previous sequences “Y” - reset sequences (start new session). If clients cannot recover the previous session they start new session with 1 and set this field to “Y”. The flag should be used with caution especially during trading session as this might lead to business data loss
553	Username		Y	Username
554	Password		Y	Password
<Trailer>			Y	

5.2. Logout (MsgType = 5)

The message initiates or confirms termination of a FIX session.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = “5”
58	Text	String (200)	N	Logout reason
<Trailer>			Y	



5.3. Resend Request (MsgType = 2)

The message is used to recover an inbound session sequence if a message was missed.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = "2"
7	BeginSeqNo	SeqNum	Y	Sequence number of the first message in range to be resent
16	EndSeqNo	SeqNum	Y	Sequence number of the last message in range to be resent
<Trailer>			Y	

5.4. Sequence Reset (MsgType = 4)

The message may be used in two modes:

- **Reset Mode** forces counterparty to adjust inbound message sequence, GapFillFlag = "N" or omitted.
- **Fill Gap Mode** is used during retransmission of messages missed by a client. Administrative messages and rejected business messages are not to be retransmitted. Instead a Sequence Reset message with GapFillFlag = "Y" must be used.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = "4"
123	GapFillFlag	Boolean	N	"N" - sequence reset, the counterparty must adjust inbound sequence number. "Y" - indicates the message is used instead of administrative or business messages which are not to be resent
36	NewSeqNo	SeqNum	Y	New adjusted sequence number
<Trailer>			Y	

5.5. Test Request (MsgType = 1)

A Test message is useful for checking sequence numbers or verifying the communication line status. Connection participant received the message is required to reply with Heartbeat message referring to TestReqID of the initial message.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = "1"
112	TestReqID	String (20)	Y	Identifier to be returned in resulting Heartbeat message
<Trailer>			Y	



5.6. Heartbeat (MsgType = 0)

Used for replying to the Test request as well as checking the status of communication.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = "0"
112	TestReqID	String (20)	Y	Identifier to be returned in resulting Heartbeat message
<Trailer>			Y	

5.7. Reject (MsgType = 3)

Issued by a party if an incoming FIX message is unsupported or not property formed. Rejected messages must not be resent if a Resend Request is received; instead a SequenceReset with GapFillFlag = "Y" is expected.

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType = "3"
45	RefSeqNum	SeqNum	Y	Sequence number of the rejected message
58	Text	String (200)	Y	Error message
373	SessionRejectReason	Int (2)	Y	Reject reason code. See Session Reject Reason Codes (tag 373).
<Trailer>			Y	



6. Drop Copy Messages

6.1. Overview

The FIX Drop Copy session sends Execution Report (MsgType = 8) messages to update the connected FIX client with order events. Customers can choose to receive messages over Drop Copy at two levels:

Trade Execution Reports including:

- Trade (ExecType=F) – Complete and partial fills
- Trade Cancel (Tag39=H) – Trade Bust message on previously published fill
- Trade Correction (Tag39=G) – Trade Correction on previously published fill

Order Status includes Trade Execution Reports plus:

- New order placed (ExecType = 0) – sent as a reply and referring to the NewOrderSingle messages to affirm the order is accepted in order book and working. The report is sent for every accepted order even for orders matched immediately.
- Order is cancelled (ExecType = 4) – sent as a reply and referring to OrderCancelRequest message to acknowledge that the cancel is accepted and the original order is no longer working. After receiving the message the original order is assumed to be cancelled, note that no separate individual execution is sent for the cancelled order itself.
- Order is replaced (ExecType = 5) – sent as a reply and referring to OrderCancelReplaceRequest messages to affirm the replace is accepted. After receiving the message the original order is assumed to be replaced, no separate individual execution is sent for the replaced order itself
- Order is rejected (ExecType = 8) – sent in case a new or cancel/replace order message is rejected
- Done for Day (ExecType = 3) – sent for all currently open orders after the trading day closes
- Expired (ExecType = C) – sent for the expired Day orders

6.2. Parties Repeating Group

Tag	Name	FIX Type	Req	Description
453	NoPartyIDs	Int	Y	Number of PartyIDs: Always 4
→ 448	PartyID	String(20)	Y	Party identifier/code
→ 447	PartyIDSource	char	Y	447 = D (Proprietary)
→ 452	PartyRole	Int	Y	1 - Subfirm code 4 - Clearing Firm Code 11 - Order Originating Trader 55 - Session ID



6.3. Execution Report: Trade (MsgType = 8, ExecType = F)

Sent by the FairX system to report order fill.

Tag	Name	FIX Type	Req	Description
<Header>				MsgType=8
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifespan of an order.
17	ExecID	String(40)	Y	Unique exchange assigned identifier of the execution
880	TrdMatchID	String(20)	Y	Match Identifier: can be used to link sides of a match
31	LastPx	Price(20)	Y	Price at which order was filled.
32	LastQty	Int(9)	Y	Quantity filled.
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
38	OrderQty	Int(9)	C	Quantity of order.
39	OrdStatus	Char(1)	Y	Indicates order status. See OrderStatus (tag 39) "1" - Partial Fill "2" - Complete Fill
40	OrdType	Char(1)	Y	Order types. See order types (Tag40)
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
44	Price	Price(20)	C	Price per single contract unit. All Execution Report messages sent in response to Market orders (with protection) and Stop orders (with protection) will include tag 44-Price populated with the Protection Price Limit (best available price +/- the protection points). If the order is not completely filled, the remaining Open Quantity will rest on the order book at the Protection Price Limit. Note: For spread trade Execution Reports, this tag is sent in the Execution Report – Fill Notice (35=8, 39=1 or 2) for the spread only and not the legs of the spread.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
59	TimeInForce	Char(1)	N	Specifies how long the order remains in effect. If not present, DAY order is the default. See Order TimeInForce (Tag59)

60	TransactTime	UTCTimeStamp(21)	Y	UTC format YYYYMMDD-HH:MM:SS.sss
75	TradeDate	LocalMktDate(8)	Y	Indicates date of trade reference in this message in YYYYMMDD format.
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150). Always "F" for Trade
151	LeavesQty	Int(9)	C	Amount of contracts remaining for execution after this fill.
393	TotalNumSecurities	Int(3)	N	Contains the number of Leg Fill Acknowledgment messages sent with the spread summary. Sent for spread fill message only.
442	MultiLegReportingType	Int(1)	N	Indicates if acknowledgment message is sent for an outright, leg of spread or spread. See MultiLegReporting (tag 442)
527	SecondaryExchangeID	String(40)	Y	Exchange assigned unique identifier; can be used to link spread summary fill notice with leg fill notice and trade cancel messages.
528	OrderCapacity	Int(1)	Y	The type of business conducted. 0 = Customer/Agency 1 = Principal
582	CustOrderCapacity	Char(1)	Y	Capacity of the customer. Used to indicate the Customer Type Indicator (CTI) Code. See CustOrderCapacity (tag582)
1028	ManualOrderIndicator	Boolean(1)	Y	Value sent on inbound message from client system indicating the order as sent manually or generated by automated trading logic.
1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)
1057	AggressorIndicator	Char(1)	C	Indicates if order was incoming or resting for the match event. See AggressorIndicator (tag 1057)
5979	RequestTime	Int(20)	N	Information carried on a response to convey the time (UTC) when the request was received by Exchange UTC format YYYYMMDD-HH:MM:SS.sss
<Trailer>				End of message.



6.4. Execution Report: Trade Cancel (MsgType = 8, Tag 39 = H)

Tag	Name	FIX Type	Req	Description
<Header>				MsgType=8
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifespan of an order.
17	ExecID	String(40)	Y	Unique exchange assigned identifier of the execution
880	TrdMatchID	String(20)	Y	Match Identifier: can be used to link sides of a match
19	ExecRefID	String(40)	Y	Contains unique ID for the trade being cancelled
31	LastPx	Price(20)	Y	Price of the canceled trade
32	LastQty	Int(9)	Y	Quantity of canceled trade
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
39	OrdStatus	Char(1)	Y	Indicates order status. See OrderStatus (tag 39) "H"=Trade Cancelled
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
60	TransactTime	UTCTimeStamp(21)	Y	UTC format YYYYMMDD-HH:MM:SS.sss
75	TradeDate	LocalMktDate(8)	Y	Indicates date of trade reference in this message in YYYYMMDD format.
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150). "H"=Trade Cancel Ack
393	TotalNumSecurities	Int(3)	N	Contains the number of Leg Fill Acknowledgment messages sent with the spread summary. Sent for spread fill message only.
442	MultiLegReportingType	Int(1)	N	Indicates if acknowledgment message is sent for an outright, leg of spread or spread. See MultiLegReporting (tag 442)
527	SecondaryExecID	String(40)	Y	Exchange assigned unique identifier; can be used to link spread summary fill notice with leg fill notice and trade cancel messages.
1028	ManualOrderIndicator	Boolean(1)	Y	Value sent on inbound message from client system indicating the order as sent manually or generated by automated trading logic.

FairX

1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)
5979	RequestTime	Int(20)	N	Information carried on a response to convey the time (UTC) when the request was received by Exchange UTC format YYYYMMDD-HH:MM:SS.sss
<Trailer>				End of message.



6.5. Execution Report: Accepted Order (MsgType = 8, ExecType = 0 or 4 or 5)

Reports with execution type New (0), Cancelled (1), Correction (2) represent an acknowledgement of successful acceptance of a New, Cancel or Cancel/Replace order.

Tag	Name	FIX Type	Req	Description
<Header>				MsgType="8"
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifespan of an order.
17	ExecID	String(40)	Y	Unique Exchange assigned execution report message identifier
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
38	OrderQty	Int(9)	Y	Order quantity. Must be a positive integer.
39	OrdStatus	Char(1)	Y	Identifies order status as accepted, cancelled or replaced. See OrdStatus (tag 39)
40	OrdType	Char(1)	Y	Order types. See order types (Tag40) Market Order types are not accepted during opening auction
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
44	Price	Price(20)	C	Required for limit or stop-limit orders. Designates the price per single contract unit.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
59	TimeInForce	Char(1)	N	Specifies how long the order remains in effect. If not present, DAY order is the default. See Order TimeInForce (Tag59) For FAK, MinQty can also be specified.
60	TransactTime	UTCTimest amp(21)	Y	Timestamp of the business event of submitting the order (e.g., click the submission button). UTC format YYYYMMDD-HH:MM:SS.sss
99	StopPx	Price(20)	C	Stop price of the order. Required for stop and stop-limit orders.
110	MinQty	Int(9)	N	Minimum quantity of an order to be executed. This tag is used only when tag 59-TimeInForce=4 (Fill and Kill).
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150)
151	LeavesQty	int(9)	Y	Amount of contract units open for further execution.

210	MaxShow	Qty(9)	N	Display quantity of an order to be shown in the order book at any given time.
378	ExecRestatementReason	Int(3)	N	Identifies origin of the order elimination. See ExecRestatementReason (tag 378)
432	ExpireDate	LocalMktDate(8)	C	Contains business date when order expires.
1028	ManualOrderIndicator	Boolean(1)	Y	N' indicates the message was generated by automated trading logic. See ManualOrderIndicator (Tag1028)
1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)
5979	RequestTime	Int(20)	N	Information carried on a response to convey the time (UTC) when the request was received by the Exchange
7928	SelfMatchPreventionID	Int(8)	C	This tag is required when market participants elect to use the optional Self Match Prevention functionality.
8000	SelfMatchPreventionStrategy	Char(1)	N	Indicates a cancel instruction when Self Match Prevention is triggered. To use tag 8000-SelfMatchPreventionStrategy, client systems must include tag 7928-SelfMatchPreventionID on the originating message See SelfMatchPreventionStrategy (tag 8000)
<Trailer>				End of message.



6.6. Execution Report: Order is Rejected (MsgType=8, ExecType = 8)

Message notifies client system of a rejected order.

Tag	Name	FIX Type	Req	Description
<Header>				MsgType=8
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifespan of an order.
17	ExecID	String(40)	Y	Unique exchange assigned identifier of the execution
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
38	OrderQty	Int(9)	C	Quantity of order.
39	OrdStatus	Char(1)	Y	Indicates order status. See OrderStatus (Tag39) "8"=Rejected
40	OrdType	Char(1)	Y	Order types. See order types (Tag40)
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
44	Price	Price(20)	C	Price per single contract unit.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
58	Text	String(200)	N	Text explanation of error code.
59	TimeInForce	Char(1)	N	Specifies how long the order remains in effect. If not present, DAY order is the default. See Order TimeInForce (Tag59)
60	TransactTime	UTCTimeStamp(21)	Y	UTC format YYYYMMDD-HH:MM:SS.sss
75	TradeDate	LocalMktDate(8)	Y	Indicates date of trade reference in this message in YYYYMMDD format.
103	OrdRejReason	Int(6)	N	Error code. See OrdRejReason (tag 103)
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150). 8=Reject Ack
151	LeavesQty	Int(9)	C	Amount of contracts remaining for execution. Always 0 for reject

FairX

1028	ManualOrderIndicator	Boolean(1)	Y	Value sent on inbound message from client system indicating the order as sent manually or generated by automated trading logic.
1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)
<Trailer>				End of message.



6.7. Execution Report: Done for Day (MsgType = 8, ExecType = 3)

Reports with execution type 'Done for Day' (3) are sent by the exchange to the firms that need the status of their open orders after the trading session is closed.

Tag	Name	FIX Type	Req	Description
<Header>				MsgType=8
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifespan of an order.
17	ExecID	String(40)	Y	Unique exchange assigned identifier of the execution
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
38	OrderQty	Int(9)	C	Quantity of order.
39	OrdStatus	Char(1)	Y	Indicates order status. See OrderStatus (Tag39) "3"=Done for Day
40	OrdType	Char(1)	Y	Order types. See order types (Tag40)
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
44	Price	Price(20)	C	Price per single contract unit.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
59	TimeInForce	Char(1)	N	Specifies how long the order remains in effect. If not present, DAY order is the default. See Order TimeInForce (Tag59)
60	TransactTime	UTCTimeStamp(21)	Y	UTC format YYYYMMDD-HH:MM:SS.sss
75	TradeDate	LocalMktDate(8)	Y	Indicates date of trade reference in this message in YYYYMMDD format.
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150). 3="Done for Day"
151	LeavesQty	Int(9)	C	Amount of contracts remaining for execution
1028	ManualOrderIndicator	Boolean(1)	Y	Value sent on inbound message from client system indicating the order as sent manually or generated by automated trading logic.
1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)

FairX

5979	RequestTime	Int(20)	N	Information carried on a response to convey the time (UTC) when the request was received by Exchange UTC format YYYYMMDD-HH:MM:SS.sss
<Trailer>				End of message.



6.8. Execution Report: Expired (MsgType = 8, ExecType = C)

Reports with execution type Expired (C) are sent by the exchange to the firms that need to track the expiration of their time-limited orders (such as DAY).

Tag	Name	FIX Type	Req	Description
<Header>			Y	MsgType="8"
1	Account	String(12)	Y	Unique account identifier.
<Parties>	Parties Group	Repeating Group	Y	see Parties Group
11	ClOrdID	String(20)	Y	Unique order identifier assigned by client system. Client system must maintain uniqueness of this value for the life of the order.
14	CumQty	Int(9)	Y	Contains cumulated traded quantity throughout lifecycle of an order.
17	ExecID	String(40)	Y	Unique exchange assigned identifier of the execution
37	OrderID	String(17)	Y	Unique exchange assigned order identifier
38	OrderQty	Int(9)	Y	Order quantity submitted by client.
39	OrdStatus	Char(1)	Y	Indicates order status. See OrdStatus (tag 39). "C" = Expired
40	OrdType	Char(1)	Y	Order types. See order types (Tag40)
41	OrigClOrdID	String(20)	N	Last accepted ClOrdID in the order chain.
44	Price	Price(20)	N	Price per unit of quantity.
54	Side	Char(1)	Y	Side of order. See Order Sides (Tag54)
55	Symbol	String(24)	Y	This tag contains the details of the instrument. Future Example: EUM20
167	SecurityType	String(6)	N	Indicates security type. See Security Type (Tag167)
59	TimeInForce	Char(1)	N	Specifies how long the order remains in effect. If not present, DAY order is the default. See Order TimeInForce (Tag59)
60	TransactTime	UTCTimest amp(21)	N	UTC format YYYYMMDD-HH:MM:SS.sss e.g. 20091216-19:21:41.109
110	MinQty	Int(9)	C	Sent in the Order Elimination message if the originating order was an FAK/FOK/MinQty that contained tag 110-MinQty.
150	ExecType	Char(1)	Y	Indicates type of execution report. See ExecType (tag 150). Always "C" = Elimination Ack
151	LeavesQty	Int(9)	Y	Quantity available for further execution. Always 0
432	ExpireDate	LocalMktD ate(8)	N	Required only if tag 59-TimeInForce=Good Till Date (GTD). Note: For spread trade Execution Reports, this tag is sent in the Execution Report – Fill Notice (35=8, 39=1 or 2) for the spread only and not the legs of the spread.

FairX

1028	ManualOrderIndicator	Boolean(1)	Y	Value sent on inbound message from client system indicating the order as sent manually or generated by automated trading logic.
1031	CustOrderHandlingInst	String(1)	Y	Defines source of original order. See CustOrderHandlingInst (tag1031)
7928	SelfMatchPreventionID	Int(8)	N	This tag is required when market participants elect to use the optional Self Match Prevention functionality.
8000	SelfMatchPreventionStrategy	Char(1)	N	This tag is required when market participants elect to use the optional Self Match Prevention functionality.
5979	RequestTime	Int(20)	N	Information carried on a response to convey the time (UTC) when the request was received by Exchange UTC format YYYYMMDD-HH:MM:SS.sss
<Trailer>				End of message.



7. Tag Values

7.1. Order Status (tag 39)

Value	Description
0	New
1	Partially filled
2	Filled
3	Done for day
4	Cancelled
5	Modify
8	Rejected
U	Undefined
C	Expired
G	TradeCorrect
H	TradeCancel

7.2. Order Types (tag 40)

Value	Description
1	Market
2	Limit
3	Stop
4	Stop Limit
5	Market Limit

7.3. Order Sides (tag 54 and tag 624)

Value	Description
1	Buy Side
2	Sell Side



7.4. Order Time-in-force (tag 59)

Value	Description
0	Day
1	Good Till Cancel (GTC)
3	Immediate Or Cancel (IOC)
4	Fill or Kill (FOK)

7.5. Position Effect (tag 77)

Value	Description
O	Open
C	Close
D	Default

7.6. Cancel Reject Reason (tag 102)

Value	Description
0	Too late to cancel
1	Unknown order

7.7. Order Reject Reason (tag 103)

Value	Description
1	Unknown symbol
2	Exchange closed
6	Duplicate order
18	Invalid price increment (submitted price precision exceeds the one supported for the instrument)
99	Other

7.8. Execution Types (tag 150)

Value	Description
0	New order Ack
F	Trade
3	Done for Day
4	Cancel Ack
5	Modify Ack
8	Reject Ack



I	Order status
H	Trade Cancel Ack
C	Elimination Ack

7.9. Security Types (tag 167)

Value	Description
FUT	Future
OPT	Option

7.10. Session Reject Reason Codes (tag 373)

Value	Description
0	Invalid tag number
1	Required tag missing
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
99	Other

7.11. ExecRestatementReason (tag 378)

Value	Description
8	Exchange
100	Cancel on Disconnect
103	Cancel Oldest (resting) due to Self Match Prevention
104	Cancel from Exchange Credit Controls violation
105	Cancel from Exchange Website
106	Cancel from Risk Management API
107	Cancel Newest (Aggressing) due to Self Match Prevention
108	Cancel due to resting order quantity less than min lot size
109	Cancel both due to Self Match Prevention

7.12. MultiLegReporting (tag 442)

Value	Description
1	Outright
2	Leg of Spread

3	Spread
---	--------

7.13. OrderCapacity (tag 528)

Identifies the origin of the order (the capacity of the firm placing the order).

Value	Description
A	Agency. Order placed by customer
P	Principal. Order placed by the firm

7.14. CustOrderCapacity (tag 582)

Customer Type Indicator as defined by the NFA

Value	Description
1	CTI 1: Transactions initiated and executed by an individual member for his/her own account, for an account he/she controls, or for an account in which he/she has ownership or financial interest.
2	CTI 2: Transactions executed for the proprietary account of a clearing member or non-clearing member firm.
3	CTI 3: Transactions where an individual member or authorized trader executes for the personal account of another individual member, for an account the other individual member controls or for an account in which the other individual member has ownership or financial interest.
4	CTI 4: Any transaction not meeting the definition of CTI 1, 2 or 3. (These should be non-member customer transactions).

7.15. CustomerOrderHandlingInst (tag 1031)

Value	Description
W	Desk
Y	Electronic (Default)
C	Vendor provided platform Billed by Executing Broker
G	Sponsored Access via exchange API
H	Premium Algorithmic trading provided by executing broker
D	Other

7.16. AggressorIndicator (tag 1057)

Value	Description
Y	Match aggressor
N	Resting at match



7.17. Manual or Automated flow (tag 1028)

Value	Description
Y	Manual
N	Automated

7.18. SelfMatchPreventionStrategy (tag 8000)

Value	Description
N	Cancel aggressing order
O	Cancel resting order
Q	Cancel both aggressing and resting orders