FIX Messages Requirements

Government – Corporate Bond and Equities

FTX Platform for Hi-MTF

Version 1.53

Highly Confidential

LEGAL NOTICES

TABLE OF CONTENTS

HISTORY OF CHANGES

Version	Date	Description
1.0	05/2007	• Initial version
1.6	04/2008	• Message "Execution Report": the possible values of a rejected order are detailed. • Message "New Order", tag 59: added "2" as possible value. • Message "Security Definition": added tag 336 as it was missing in the documentation.
1.7	08/2008	• Message "Execution Report": added the last six values to the possible values of a rejected order
1.8	09/2008	• Message "Security Definition": • Added group NoTicks (tags 5600, 5601, 5602, 5563), page 53 • Tag 5563 now belongs to group NoTicks, page 53 • Added tag 64, page 53 • Message "Security Status": added tag 336, page 55
1.9	03/2009	• Message "New Order": tag 1 became optional and the string length became 16, see page 18 • Message "Execution Report": tag 1 became optional and the string length became 16, see page 26
1.10	08/2009	• Message "Market Data Snapshot/Full Refresh": the field "Security ID" became optional, see page 56
1.11	10/2009	• Message "Execution Report": added tag 19 'ExecRefID', see page 26 • Message "Security Definition": added value 5 = EQUITY to tag 460, see page 53
1.12	02/2010	• In various Messages: added details about floating-point representation, see page 73
1.13	03/2010	• Message "Market Data - Snapshot /Full Refresh": added value for tag 269, page 56 and page 65
1.14	04/2010	• Message "Execution Report": added value for tag 452 (see 3.2)
1.15	03/2011	• Message "Market Data - Snapshot /Full Refresh": modified value '4' of the tag 269 and added tag 271 for value '4', '5' and 'Q' of the tag 269, see page 56 and page 65
1.16	05/2011	• Message "Security Definition": modified the comment of both tag 55 and 48, page 53
1.17	7/2011	• Message "Security Definition": added tag 58, see page 53

1.18	02/2013	• Message "Execution Report": added tag 48 ,tag 22 and tag 7665, page 26 • Message "Market Data - Snapshot /Full Refresh": added tag 262, page 56
1.19	11/2013	• Message "Logon": added some details about the tags that are used
1.20	04/2014	• Message "New Order": added tag 376, page 18 • Message "Security Status Request": modified the comment/value of both tag 48 and tag 22, page51 • Message "Security Definition", "Security Status" and "Market Data – Snapshot / Full Refresh": modified the comment/value of the tag 48, tag 22 and tag 55, page 53, page 55 and page 56 • Note: a modification of string length is indicated by an asterisk next to the number in the *Data Type* column
1.21	10/2014	• Message "Execution Report": Updated the Possible errors of a rejected Order
1.22	10/2014	• Message "Security Definition": added the value '12 = OTHER' to the tag 460, page 53
1.23	02/2015	• Message "Market Data - Snapshot /Full Refresh": added tag 270 for value 'Trade Volume' of the tag 269, page 63 • Message "Market Data - Snapshot / Full Refresh Sent following every Trade": added tag 269, 270 and 271 (tag 269: 'Volume Trade') page 70
1.24	10/2015	• FIX versions: added note on RFCQ buy-side trading supported version. • Section 3: Trading Application Level messages: the message summary table has been updated and extended with messages related to RFCQ trading • Message "Execution Report" (Section 5.2, page 26): added QuoteRespID, QuoteID, QuoteMsgID fields, added a value to ExecType (150) field and changed some field descriptions to include the RFCQ workflow • Added Section 3.3 RFCQ buy-side trading
1.25	10/2015	• Message "Execution Report": added tag 120, tag 155 and tag 156, page 34
1.26	12/2015	• Message "Quote Request": added tag 528, tag 600 and deleted tag 686, page 39 • Message "Quote Response": added tag 9770 and tag 600, page 42
1.27	03/2017	• Message "Execution Report": added tag 9730, page 34
1.28	04/2017	• Message "Execution Report": changed values for tag 9730, page 34

1.29	05/2017	• Fix versions: compatibility with versions earlier than 4.4 was removed • Section Message Header: tag 8 begin string FTX 4.4 and tag 52 in milliseconds, page 13 • Section New Order Type: tag 59 added Good till time for 6 value, tag 126 ExpirationTime when time in force is 6 in microseconds, tag 528 OrderCapacity added R= Riskless principal, tag 529 added note, tag 58: added note, tag 9215 added. For the tags 543, 448, 447, 452 removed the note: Used only for 4.3 and 4.4. For the tag 447, added M and N. For the tag 448, added a note. Tag 2594 and 2595 added. Tag 376 removed. Tag 60 in microseconds, string from 17 to 24. See page 18 • Section Order Cancel Reject: tag 60 in microseconds, string from 17 to 24. • Message "Execution Report": tag 150, removed values 1 and 2; tag 20, removed value 3; tag 39, removed value 5; tag 528 OrderCapacity added R= Riskless principal; tag 126 and 9215 added; tag 7665 removed; tag 9730 added value C=Auction; tag 60 in microseconds, string from 17 to 24. See page 26 • Possible errors for rejection order added: Invalid Settlement Date, Cannot update, too early, Invalid RFQ action, Invalid RFQ Quote Image, RFQ Invalid Yield, Attention: Accept Not Allowed, There is already an exception. Please restore first, Planning section time earlier than security planning time, Invalid OTC Status, Not Enabled Member, Not Allowed AON, Qty Higher Than Issue Qty, Too many transactions per time unit. See page 26 • Section Quote Request (Type R): tag 60 in microseconds, string from 17 to 24; tag 448, added note; tag 447, added M and N; tag 452, removed 1 and added 3, 5 and 12 • Section Quote Request Reject: tag 447, added M and N. • Section Quote Type (S): tag 447, added M and N. • Section Quote Response (Type AJ): tag 452, removed 1 and added 3, 5 and 12, tag 62, 126 and 60: time stamps in microseconds, string from 17 to 24. • Section Security Definition: tag 106 string 64 instead of 16; tags added: 461, 454, 455, 456. See page 51 • Section Market Data Snapshot full Refresh (type W) used as a response: tag 273 in microseconds. See page 56 • Section Market Data - Snapshot / Full Refresh Sent following every Trade: tag 273 in microseconds. See page 56 • Section Market Data - Snapshot / Full Refresh Sent when there is a change in the Order Book: tag 273 in microseconds. See page 56

1.30	05/2017	• Execution Report: added tags 1430, 625, 574, 828, 829, 1839, 1115, 2668, 2669, 2670, 855, 2405, 2667, 1390, 1934 (to map MMT Flags). See page 26 • Market Data Snapshot full Refresh (MDEntryType =2) and Market Data Snapshot Sent following every Trade: added tags 1034, 5463, 625, 574, 828, 829, 1838, 1839, 1115, 2668, 2669, 2670, 277, 2405, 2667, 1390, 1934 (to map MMT Flags). See page 56 and 65
1.31	08/2017	• Market Data Snapshot full Refresh (type W): tag 273 (the second one), string from 12 to 15. See page 56 • Security Status (type f): re-inserted tag 336. See page 55
1.32	08/2017	• Removed asterisk added in version 1.20 (see the note) • Tag 9215 mark as not used in HiMTF Exchange.
1.33	08/2017	• Corrected tag number for MDOriginType from 1034 to 1024 (Page 50 and 56)
1.34	05/2018	• Updated the possible values for the tag 268 NoMDEntries (page 63).
1.35	06/2018	• Review of the Market Data - Snapshot / Full Refresh Sent when there is a change in the Order Book (page 70) and of the tag 268 (page 63)
1.36	10/2018	• New Order (Type D) and Quote Request (Type R) messages: Updated the description of tag 448
1.37	10/2018	• Message f: added the value “CRB” to tag 336
1.38	11/2019	• General Revision template • Execution Report: Removed unused value 5, C and I of tag 150. Removed tag 20 that is deprecated. Added value A to tag 39.
1.39	06/2020	• New template applied • Removed tag 9215 unused by Hi-MTF • Message W (refresh and refresh for a book change): ◦ added tag 269 with custom the value b and the corresponding tag 270, 271, 272,273,37,293,4002,453,448,447 and 452 ◦ added tag 269 with custom the value o and the corresponding tag 270, 271, 272,273,37,293,4002,453,448,447 and 452
1.40	07/2020	• added tag 4003 to message 8
1.41	03/2021	• “MLQ” value added to tag 336 in message f)
1.42	12/2021	• Added "MGT" = Management to tag 336
1.43	1/2022	• Tag 115 in Standard header becomes mandatory
1.44	3/2022	• Tag 4000 added to Security Definition (type d)

1.45	04/2022	• Tag 267 and tag 146 become "not mandatory"
1.46	07/2022	• Quote Request (Type R): Tag 7252 and tag 7254: "Zero is an admitted value" added • Quote (Type S): Tag 7252 and tag 7254, tag 647 and tag 648: "Zero is an admitted value" added • Market Data - Snapshot / Full Refresh Sent following every Trade: tag 262 always "dummyid"
1.47	08/2022	• Tag 2594 and 2595 added to Execution Report (Type 8)
1.48	08/2022	• Tag 311 added to Security Definition (type d) message
1.49	12/2022	• Tag 271: new definition from "Quantity traded at the Closing Price" to "Total volume (quantity) traded" when MDEntryType=5
1.50	08/2023	• Tag 4001 added to market Data Request (type V)
1.51	12/2023	• Order Cancel/Replace Workflow schema split to display the two outcomes of the transaction
1.52	01/2024	• Order Cancel/Replace (Type G) Workflow – Successful Transaction: Execution type "New" replaced with Execution type "Replaced" • Tag 150: E and 5 values added • Order Cancel (Type F): tag 432 added
1.53	01/2024	• Order Cancel (Type F): tag 126 added

1 INTRODUCTION

1.1 FTX PLATFORM OVERVIEW

An FTX platform is a solution for running an electronic market: financial institutions who wish to create an MTF or regulated markets to improve the trading software.

Both types of platforms supports Government Bonds, Corporate Bonds and Equities.

1.2 PURPOSE

The document describes the specifications for communicating with an FTX platform via FIX protocol. FIX ("Financial Information eXchange Protocol") is a messaging standard developed specifically for the real-time electronic exchange of securities transactions.

It is a public-domain specification owned and maintained by FIX Protocol, Ltd. More information about the FIX protocol may be found at http://www.fixprotocol.org.

Clients can submit orders to be staged on an FTX platform with a standardized method via FIX.

This document covers the FIX messages and fields that are supported by an FTX platform, describing the message flows and the requirements for integrating with an FTX platform using FIX. When there are no specifications, clients should refer to the standard FIX protocol.

FTX platform ignores any tags that are not specified in this document (i.e. FTX platform does not manage these tags) whenever such tags are in the header of the messages or in the messages itself.

1.3 FIX VERSIONS

The only FIX version available is 4.4. Transactions sent with lower versions will be rejected.

1.4 FIX SESSION

The Client application is supposed to establish two different fix sessions to two different fix servers for trading and market data messages.

1.5 DOCUMENT CONVENTIONS

Each message is represented as a table, where each row is a message field or component block. The following characteristics are described for each field:

- **Tag**: unique field identifier
- **Field** Name: field name
- **Content**: list of the valid values and additional information
- **Data Type**: field type
- **Req**: indicates whether the field is required or not in appropriate message or component block. The possible values are:
    - '**Y**': tag is required (mandatory)
    - '**N**': tag is not required (optional)
    - '**C**': tag is conditionally required

2 SESSION MESSAGES

The following sections outline the standard tags used in the supported message types.

The following convention is used in this document to indicate message direction:

- In: a message sent to the FTX platform
- Out: a message sent by the FTX platform
- In/Out: a message that can be sent to or from the FTX platform

The session messages are the following:

Message In: Logon, Logout, Test Request, Heartbeat, Resend Request and Sequence Reset.

Message Out: Logon, Logout, Test Request, Heartbeat, Session Reject.

2.1 STANDARD HEADER

All messages contain a standard set of header fields, described below.

Tag	Field Name	Req	Comments
8	BeginString	Y	Always the first field of the message and set to: FIX 4.4.
9	BodyLength	Y	Message length in bytes. Always the second field of the message.
35	MsgType	Y	Message type. Always the third field of the message. • D = NewOrderSingle • F = OrderCancelRequest • G = OrderCancelReplaceRequest • 9 = OrderCancelReject • R = Quote Request • AJ = Quote Response • AG = Quote Request Reject • S = Quote • AI = Quote Status Report • 8 = Execution Report • c = Security Definition Request • e = Security Status Request • V = market Data Request • d = Security Definition • f = Security Status • W = Market Data – Snapshot / Full Refresh
34	MsgSeqNum	Y	Message sequence number.
49	SenderCompID	Y	Assigned value used to identify message sender.
115	OnBehalfOfCompID	Y	Sender Member ID.
56	TargetCompID	Y	TargetCompID For incoming orders.
52	SendingTime	Y	Time of message transmission. Format: YYYYMMDD-hh:mm:ss.mmm

43	PossDupFlag	N	Indicates possible retransmission of message with this sequence number. • N = Original transmission • Y = Possible duplicate Not Applicable for Market Data Messages.
97	PossResend	N	Indicates that the message may contain information that has been sent under another sequence number. • N = Original transmission • Y = Possible resend. Not Applicable for Market Data Messages.

2.2 STANDARD TRAILER

All messages contain a standard trailer field, described below.

Tag	Field Name	Req	Comments
10	CheckSum	Y	Per FIX Specification

2.3 LOGON (IN)

This message is sent to initiate a FIX session to the FTX Platform. The Logon message establishes the communication session, authenticates the client connecting, and initializes the message sequence number.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = A
98	EncryptMethod	Y	Method of encryption
108	HeartBtInt	Y	Heartbeat interval (seconds)
141	ResetSeqNumFlag		Indicates that both sides of the FIX session should reset sequence numbers.
553	Username	Y	The username must be set as follows: operator@member
554	Password		Password
	Strandard Trailer	Y	

2.4 HEARTBEAT (IN/OUT)

This message sent by The FTX Platform during periods of application inactivity to ensure connection validity. The receiving party should always respond with a heartbeat message.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = 0
112	TestReqID		Identifier included in Test Request message to be returned in resulting Heartbeat
	Strandard Trailer	Y	

2.5 RESEND REQUEST (IN/OUT)

It is a request that certain messages be resent. Often used when gaps detected in the sequence numbering, when a message is lost, or during the initialization process.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = 2
7	BeginSeqNo	Y	Message sequence number of first message in range to be resent
16	EndSeqNo	Y	Message sequence number of last message in range to be resent. If request is for a single message BeginSeqNo (7) = EndSeqNo. If request is for all messages subsequent to a particular message, EndSeqNo = "0" (representing infinity).
	Strandard Trailer	Y	

2.6 SEQUENCE RESET

This message is used to skip administrative messages on resend and to reset sequence on client request.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = 4
123	GapFillFlag	N	Indicates that the Sequence Reset message is replacing administrative or application messages which will not be resent. • Y = Gap Fill message, MsgSeqNum field valid • N = Sequence Reset, ignore MsgSeqNum
36	NewSeqNo	Y	New sequence number
	Strandard Trailer	Y	

2.7 TEST REQUEST (IN/OUT)

This message is used to verify session connectivity and to synchronize sequence numbers. The receiving party should always respond with a heartbeat message.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = 1
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat
	Strandard Trailer	Y	

2.8 LOGOUT (IN)

This message signals the normal termination of a trading session. A session terminated without a Logout message will be considered an abnormal condition.

Tag	Field Name	Req	Comments
	Standard Header	Y	MsgType tag 35 = 5
58	Text		MsgType tag 35 = 1
	Strandard Trailer	Y	

3 TRADING APPLICATION LEVEL MESSAGES

The following application level messages are currently supported.

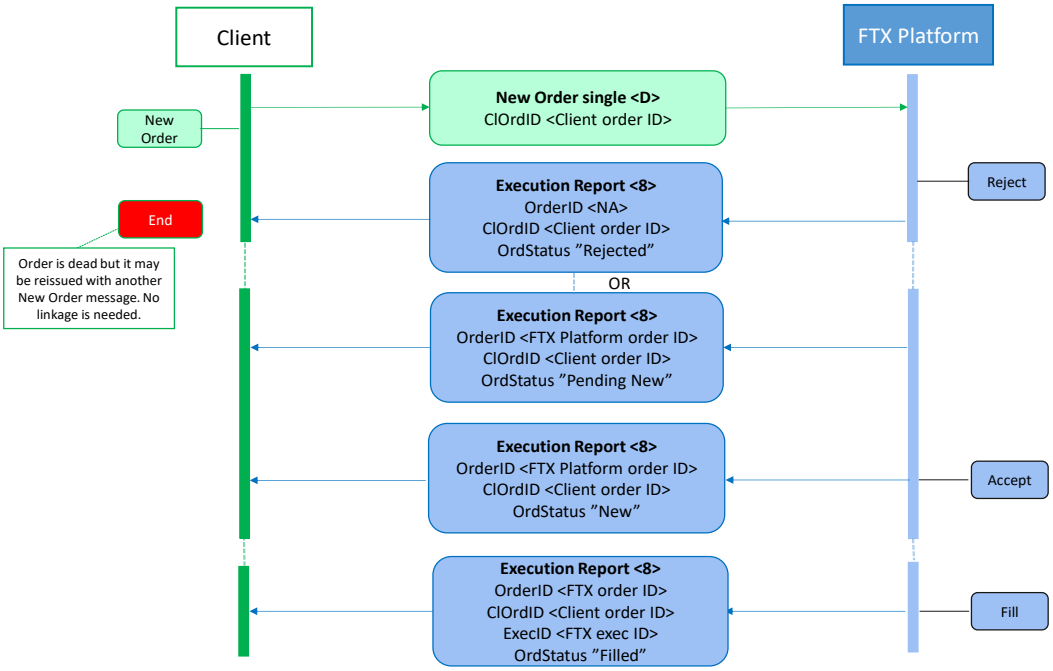
Type	Name	Direction	Description
8	Execution Report	OUT	Used to convey order status information and Execution information
9	Order Cancel Reject	OUT	Used to reject a Cancel request or Cancel/replace request message that cannot be fulfilled
D	New Order	IN	Used to submit a new order
F	Order Cancel Request	IN	Used to request the cancellation of the whole remaining quantity of an existing order
G	Order Cancel/Replace	IN	Used to change the parameters of an existing order
R	Quote Request	IN	Used to submit a new RFCQ or to request the cancellation of the whole remaining quantity of an existing RFCQ [RFCQ buy-side workflow]
AG	Quote Request Reject	OUT	Used to reject a new RFCQ or a Cancel request that cannot be fulfilled [RFCQ buy-side workflow]
S	Quote	OUT	Used to convey Dealer's quotes [RFCQ buy-side workflow]
AJ	Quote Response	IN	Used to hit/lift Dealer's quote or to reject it
AI	Quote Status Report	OUT	Used either to acknowledge a Quote Request (new RFCQ or Cancellation Request) or to reject a Quote Response, or unsolicited to notify status changes of quotes.

3.1 ORDER ENTRY

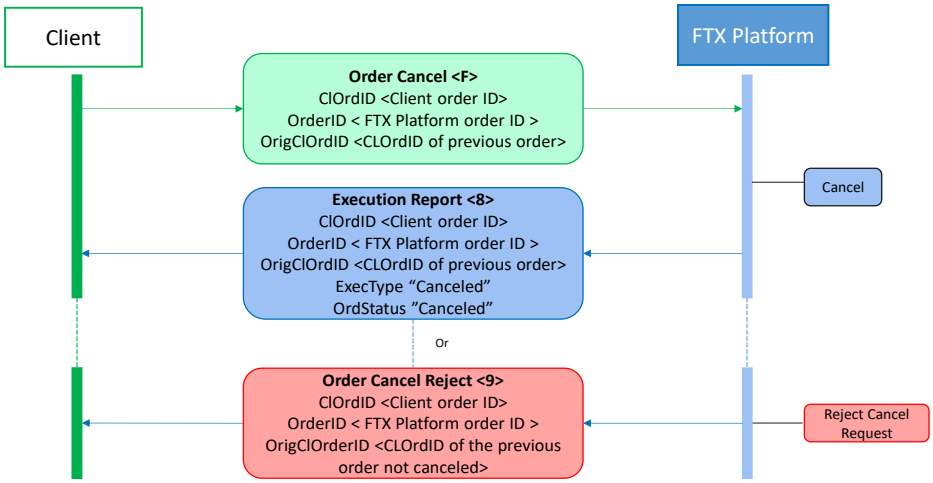
This subsection details the workflows and the messages for the order negotiations.

3.1.1 Order workflow

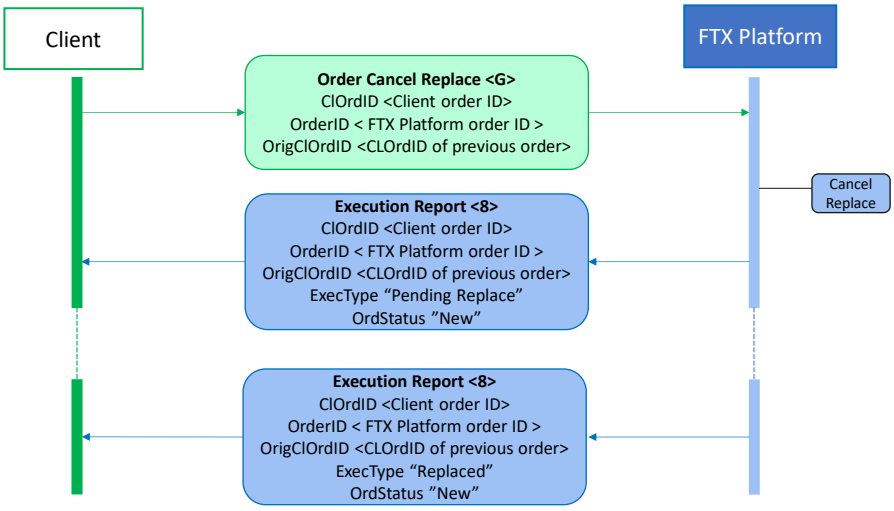
New Order



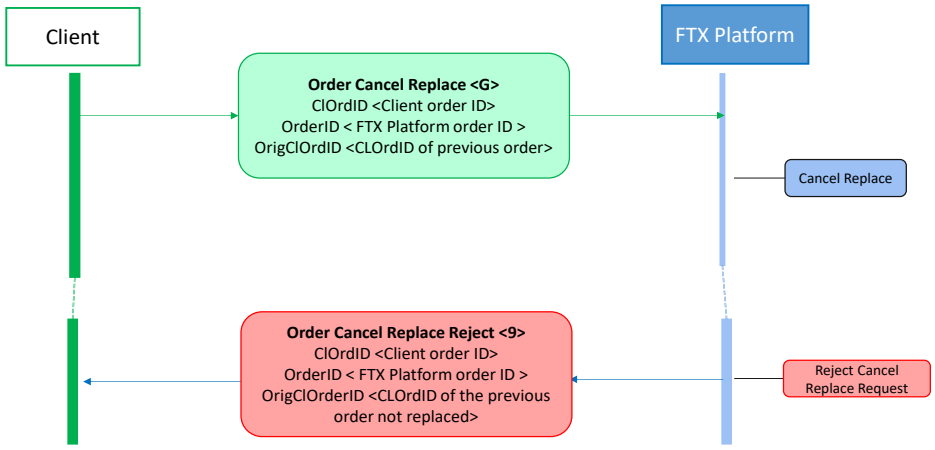
Order Cancel (Type F) Workflow



Order Cancel/Replace (Type G) Workflow – Successful Transaction



Order Cancel/Replace (Type G) Workflow – Aborted Transaction



3.1.2 New Order (Type D)

Initiation of a new order by the client.

For the fields with a "String" data type, a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared. The message will not be rejected.

Tag	Field Name	Content	Data Type	Req
<Standard Header>		MsgType <35> = D		Y
11	ClOrdID	Unique identifier assigned by the client	String (32)	Y
40	OrdType	Valid values: • 1 = Market • 2 = Limit	Char	Y
54	Side	Valid values: • 1 = Buy • 2 = Sell	Char	Y
100	ExDestination	Execution destination.	String (24)	Y
Component Block<Instrument>				
48	SecurityID	Security identifier value of SecurityIDSource (22) type, valid value ISIN. Requires SecurityIDSource.	String (12)	Y
22	SecurityIDSource	• 4 = ISIN number	String (1)	Y
End Component Block<Instrument>				
44	Price	Price per contract	Float	Y
423	PriceType	Code to represent the price type. Valid values: • 1 = Percentage (Default) • 9 = Yield	Char	N
Component Block<OrderQtyData>				
38	OrderQty	Number of contracts	Integer	Y
End Component Block<OrderQtyData>				
110	MinQty	Minimum quantity of an order to be executed.	Integer	N
111	MaxFloor	Maximum quantity (e.g. number of shares) within an order to be shown on the Exchange floor at any given time.	Integer	N

1	Account	Account for which the contracts are to be bought or sold.	String (32)	N
59	TimeInForce	Valid values: • 0 = Day • 1 = Good Till Cancel (GTC) • 2= At the Opening (OPG) The order is valid until the next auction phase • 3 = Immediate or Cancel (IOC) • 4 = Fill or Kill (FOK) • 6 = Good Till Date or Good Till Time	Char	N
432	ExpireDate	Date the order expires when it is of Good Till Date type (59=6) Format: YYYYMMDD	String (8)	N
126	ExpireTime	ExpirationTime when time in force is 6. YYYYMMDD-hh:mm:ss.μμμμμμ If this tag is specified, tag 432 will be ignored	String (24)	N
528	OrderCapacity	Valid values: • G = Proprietary • I = Individual (Client) • R= Riskless principal. It should be used only for Matched Principal activities.	Char	Y
529	OrderRestrictions	Valid values: • 5 = Acting as Market Maker or Specialist in the security. It can be used only if OrderCapacity is G (Proprietary)	String (1)	N
58	Text	Free Text	String (64)	N
Component Block<Parties>				
453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries.	Integer	N
->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452). The PartyID can contain a generic string only if PartyRole = 3 (ClientID). In this case, PartyIDSource must be D (Custom Code). For the other values of PartyIDSource, the PartyID must contain a number (number 0,1,2,3 are currently reserved for NoClient, ALGO, PNAL and CLIENT). Note that if PartyRole = 3 or 5, the Party Identifier must be between 0 and 4294967295	String (32)	N
->447	PartyIDSource	Identifies class or source of the PartyID (448) value • D = Proprietary/Custom code • M = Algorithm Short Code • N = Natural Person Short Code Requested for MiFID II requirements.	Char	N
->452	PartyRole	Identifies the type or role of the PartyID Valid Value • 3 = ClientID • 5 = Investor ID (the Investment Decision Maker) • 12 = Executing Trader (the Executing Decision Maker)	Integer	N
End Component Block<Parties>				

60	TransactTime	Time this order request was initiated/released by the trader, trading system, or intermediary. Timestamp with following format: YYYYMMDD-hh:mm:ss.μμμμμμ E.g. 20170620-10:06:51.453765	String (24)	Y
2594	OrderAttributeType	Indicates if the order was sent to reduce risk in an objectively measurable way in accordance with Article 57 of Directive 2014/65/EU • Only possible value: R	Char	N
2595	OrderAttributeValue	Mandatory if OrderAttributeType is specified. • The value must be Y	Char	N
<Standard Trailer>				Y

3.1.3 Order Cancel (Type F) / Cancel/Replace (Type G)

Client request to cancel-replace a pre-existing order.

For the fields with a data type "String", a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared. The message will not be rejected.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = F or G		Y
11	ClOrdID	Unique identifier assigned by the client	String (32)	Y
100	ExDestination	Execution destination.	String (24)	Y
Component Block<Instrument>				
48	SecurityID	Security identifier value of SecurityIDSource (22) Requires SecurityIDSource.	String (12)	Y
22	SecurityIDSource	• 4 = ISIN number	Char	Y
Component Block<Instrument>				
54	Side	Valid values: • 1 = Buy • 2 = Sell	Char	Y
37	OrderID	Exchange order id	String (32)	Y
41	OrigClOrdID	CLOrdID of previous order	String (32)	Y
44	Price	Price Used only in Order Cancel/Replace Request (type G)	Float	N
Component Block< OrderQtyData>				
38	OrderQty	Quantity ordered	Qty	N
End Component Block< OrderQtyData>				
60	TransactTime	Time this order request was initiated/released by the trader, trading system, or intermediary. Timestamp with following format: YYYYMMDD-hh:mm:ss.μμμμμμ E.g. 20170620-10:06:51.453765	String (24)	Y
432	ExpireDate	Date the order expires when it is of Good Till Date type (59=6) Format: YYYYMMDD	String (8)	N

126	ExpireTime	ExpirationTime when time in force is 6. YYYYMMDD-hh:mm:ss.μμμμμμ If this tag is specified, tag 432 will be ignored	String (24)	N
	<Standard Trailer>			Y

3.1.4 Order Cancel Reject (Type 9)

FTX platform sends a rejection of a new/cancel/replace order request from the client.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = 9		Y
11	ClOrdID	Unique identifier assigned by the client	String (32)	Y
41	OrigClOrderID	It contains the ClientOrderID that has not been canceled/replaced. Please note that the ClientOrderID found in this tag is the last accepted order (NOT the initial order of the day)	String (32)	Y
39	OrdStatus	OrdStatus value after the CancelReject is applied	Char	Y
37	OrderID	Exchange order id	String (32)	Y
434	CxlRejResponseTo	Identifies the type of request that a Cancel Reject is in response to. Valid values: • 1 - Order Cancel Request • 2 - Order Cancel/Replace Request	Char	Y
102	CxlRejReason	Code to identify reason for order rejection Valid Values: • 0 = Too late to cancel • 1 = Unknown order • 3 = Order already in Pending Cancel or Pending Replace status • 6 = Duplicate ClOrdID (11) received • 99 = Other	Integer	N
58	Text	Error String	String (64)	N
	<Standard Trailer>			Y

3.2 EXECUTION REPORT (TYPE 8)

Execution reports sent back to the client.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = 8		Y
11	ClOrdID	Unique identifier assigned by the client to the original request (order or RFCQ)	String (32)	Y
41	OrigClOrdID	CLOrdID of previous order. Required when the message is sent after order modification / cancellation	String (32)	C
693	QuoteRespID	Unique identifier for Quote Response message. Required if responding to a QuoteResponse message (RFCQ workflow only).	String (32)	C
117	QuoteID	Unique identifier for the quote that generated the deal (RFCQ workflow only).	String (32)	C
1166	QuoteMsgID	Unique identifier for the quote message that generated the deal (RFCQ workflow only).	String (40)	C
1	Account	Account for which the contracts are to be bought or sold	String (32)	N
54	Side	Valid values: • 1 = Buy • 2 = Sell	Char	Y
Component Block< OrderQtyData>				
38	OrderQty	Ordered quantity (Number of lots)	Integer	Y
End Component Block< OrderQtyData>				
44	Price	Price	Float	Y
423	PriceType	Code to represent the price type. Valid values: • 1 = Percentage (default) • 9 = Yield	Char	N
40	OrdType	Valid values: • 1 = Market • 2 = Limit	Char	Y
37	OrderID	Order ID to which current execution report refers. In case of RFCQ workflow, it contains the unique identifier assigned by the market to the original quote request.	String (32)	Y

150	ExecType	Describes the purpose of the specific Execution Report (i.e. Pending Cancel). Note that the OrdStatus will always identify the current order status. Valid values: • 0 = New • 4 = Canceled • 5 = Replaced • 6 = Pending Cancel • 8 = Rejected • A = Pending New • E = Pending replace • F = Trade (partial fill or fill) • G = Trade Correct (formerly an ExecTransType) • H = Trade Cancel (formerly an ExecTransType)	Char	Y
4003	SettlMode	Code to identify if the market sends the trade to a settlement system. The value in this field should only be considered if the ExecType is set to (F) or (G) or (H) Valid Values: • M: Manual • A: Automatic	Char	C
39	OrdStatus	Valid values: • 0 = New • 1 = Partially filled • 2 = Filled • 4 = Canceled • 6 = Pending Cancel (e.g. result of Order Cancel Request) • 8 = Rejected • A= Pending New	Char	Y
17	ExecID	Unique identifier of execution message as assigned by the marketplace	String (32)	Y
31	LastPx	Price of last fill	Float	N
32	LastQty	Quantity of last fill	Qty	N
6	AvgPx	Current average price of all fills on current order. This will always be set to 0.	Float	Y
14	CumQty	Total executed contracts on the current order.	Integer	Y
151	LeavesQty	Remaining contracts to be executed on the current order.	Integer	Y
432	ExpireDate	Date order expires when order type is Good Till Date (59=6). Format: YYYYMMDD	String (8)	N
126	ExpireTime	ExpirationTime, taken from the order. Format: YYYYMMDD-hh:mm:ss.μμμμμμ	String (24)	N

528	OrderCapacity	Valid values: • G = Proprietary • I = Individual (Client) • R= Riskless principal. It is used only for Matched Principal activities	Char	Y
529	OrderRestrictions	Valid values: • 5 = Acting as Market Maker or Specialist in the security	String (1)	N
Component Block<Parties>				
453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries. Possible parties entries: • 1-First Party: is the member • 2-Second Party: is the member operator. • 3-Third Party: is the counterpart • 4-Fourth Party: is the counterpart capacity	Integer	N
->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452). For PartyRole=37, possible values are "C" or "P". "C" is used for third accounts and "P" for Proprietary.	String (16)	N
->447	PartyIDSource	Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Note: applicable values depend upon PartyRole (452) specified. • D = Proprietary/Custom code • M = Algorithm Short Code • N = Natural Person Short Code	Char	N
->452	PartyRole	Identifies the type or role of the PartyID (448) specified. Valid Value • 1 = Executing Firm • 12 = Executing Trader (associated with Executing Firm - actually executes) • 17 = Contra Firm • 37 = Contra Trader, used to receive counterpart capacity	Integer	N
End Component Block<Parties>				
64	SettlDate	Specific date of trade settlement (SettlementDate) in YYYYMMDD format	String (8)	Y
103	OrdRejReason	Code to identify reason for order rejection Valid Values: • 1 = Unknown symbol • 4 = Too late to enter • 6 = Duplicate Order (e.g. dupe ClOrdID (11)) • 99 = Other	Integer	N
58	Text	Contains the free text entered by the user or error if order status = 8 For more details see Appendix	String (64)	N

60	TransactTime	Time of execution/order creation Timestamp with following format: YYYYMMDD-hh:mm:ss.μμμμμμ E.g. 20170620-10:06:51.453765	String (24)	Y
19	ExecRefID	If a contract has been canceled, it contains the identifier of the canceled contract	String (32)	N
Component Block<Instrument>				
48	SecurityID	Security identifier value of SecurityIDSource (22) type, valid value ISIN. Requires SecurityIDSource.	String (12)	Y
22	SecurityIDSource	• 4 = ISIN number	Char	Y
End Component Block<Instrument>				
120	SettlCurrency	Settlement Currency	String (3)	Y
155	SettlCurrFxRate	Foreign exchange rate used to compute SettlCurrAmt from Currency to SettlCurrency	Float	Y
156	SettlCurrFxRateCalc	Specifies whether the SettlCurrFxRate should be multiplied (M) or divided (D)	Char	Y
9730	TradeLiquidityIndicator	Indicates whether the order added or removed liquidity. The value in this field should only be considered if the ExecType is Trade (F). Valid values: • A: Provider, Added Liquidity • R: Aggressor, Removed Liquidity • C: Auction	Char	N
1430	VenueType	This tag maps the MMT level 1: "Market Mechanism". Possible values are: • B = Central limit order book • Q = Quote driven market • D = Dark order book • O = Off-market • A = Auction driven market • N = Quote negotiation • H = Hybrid Market The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Char	N

625	TradingSessionSubID	This tag, together with tag 574, maps the MMT level 2: "Trading Mode". Possible values are: • 8 = Any auction • 2 = Opening or opening auction • 4 = Closing or closing auction • 6 = Intraday auction • 9 = Unscheduled intraday auction • 3 = (Continuous) trading • 5 = Post-trading • 10 = Out of main session trading The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N
574	MatchType	This tag, together with tag 625, maps the MMT Level 2 "Trading Mode". Possible values are: • 3 = Confirmed trade report (reporting from recognized markets) • 1 = One Party Trade Report (privately negotiated trade) • 9 = Systematic Internalizer The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N
828	TrdType	This tag, together with tag 1839, maps the MMT Level 3.1 "Transaction Type, Transaction Category". Possible values are: • 62 = Dark trade • 65 = Package trade • 2 = Exchange for physical The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
829	TrdSubType	This tag maps the MMT level 3.3: "Transaction Type: Agency Cross Indicator". Possible values are: • 37 = Crossed trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
1838	NoTradePriceConditions	Number of TradePriceCondition (1839) entries	Integer	N

1839	TradePriceCondition	This tag, together with tag 828, maps the MMT Level 3.1 "Transaction Type, Transaction Category", MMT Level 3.6 "Special Dividend", MMT Level 3.8 "Transaction Type: Ordinary Trades or Trades outside price formation". Possible values are: • 13= Special Dividend • 14 = Price improvement • 15 = Non-price forming trade • 16 = Trade exempt from trading obligation • 17 = Price is pending The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
1115	OrderCategory	This tag, together with tags 2669 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". Possible values are • 3 = Privately negotiated trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Char	N
Component Block <TrdRegPublicationGrp>				
2668	NoTrdRegPublications	Number of TrdRegPublicationType (2669) and TrdRegPublicationReason (2670) entries.	Integer	N
->2669	TrdRegPublicationType	This tag, together with tags 1115 and tag 2669, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are • 0 = Pre-trade transparency waiver • 1 = Post-trade deferral The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N

->2670	TrdRegPublicationReason	This tag, together with tags 1115 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are: • 1 = No preceding order in book as transaction price depends on system-set reference price for an illiquid instrument) • 0 = No preceding order in book as transaction price set within average spread of a liquid instrument) • 2 = No preceding order in book as transaction price is subject to conditions other than current market price • 3 = No public price for preceding order as public reference price was used for matching orders • 4 = No public price quoted as instrument is illiquid • 5 = No public price quoted as order size is above standard market size The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
End Component Block <TrdRegPublicationGrp>				
855	SecondaryTrdType	This tag, together with tags 2669 and 2670, maps MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator". Possible values are • 64= Benchmark The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
2405	ExecMethod	This tag maps Level 3.7 "Transaction Type: Off Book Automated Indicator". Possible values are • 0 = Undefined/unspecified • 1 = Manual (the transaction was executed in a manual or other non-automated manner) • 2 = Automated (the transaction was executed on an automated execution platform such as an automated systematic internalizer system, broker crossing network, dark pool trading, "direct to capital" systems, broker position unwind mechanisms, etc.) The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N

2667	AlgorithmicTradeIndicator	This tag, together with tags 2669 and tag 2670, maps MMT Level 3.9 "Transaction Type: algorithmic indicator". Possible values are • 1 = Algorithmic trade • 0 = Non-algorithmic trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
1390	TradePublishIndicator	This tag maps MMT Level 4.1 "Publication Mode and Post Trade Deferral". Possible values are • 1 = Publish trade • 2 = Deferred publication The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
1934	RegulatoryReportType	This tag maps MMT Level 4.2 "Publication Mode and Post Trade Deferral". Possible values are • 11 = Limited details trade • 12 = Daily aggregated trade • 13 = Volume omission trade • 14 = Four weeks aggregation trade • 15 = Indefinite aggregation trade • 16 = Volume omission trade, eligible for subsequent enrichment in aggregated form • 17 = Full details of earlier "limited details trade" • 18 = Full details of earlier "daily aggregated trade" • 19 = Full details of earlier "volume omission trade" • 20 = Full details of earlier "four weeks aggregation trade" • 21 = Full details of earlier "volume omission trade, eligible for subsequent enrichment in aggregated form" The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
2594	OrderAttributeType	Indicates if the order was sent to reduce risk in an objectively measurable way in accordance with Article 57 of Directive 2014/65/EU Only possible value: R	Char	N
2595	OrderAttributeValue	Mandatory if OrderAttributeType is specified. Only possible value: Y	Char	N
	<Standard Trailer>			Y

3.3 REQUEST FOR COMPETITIVE QUOTE - BUY-SIDE TRADING

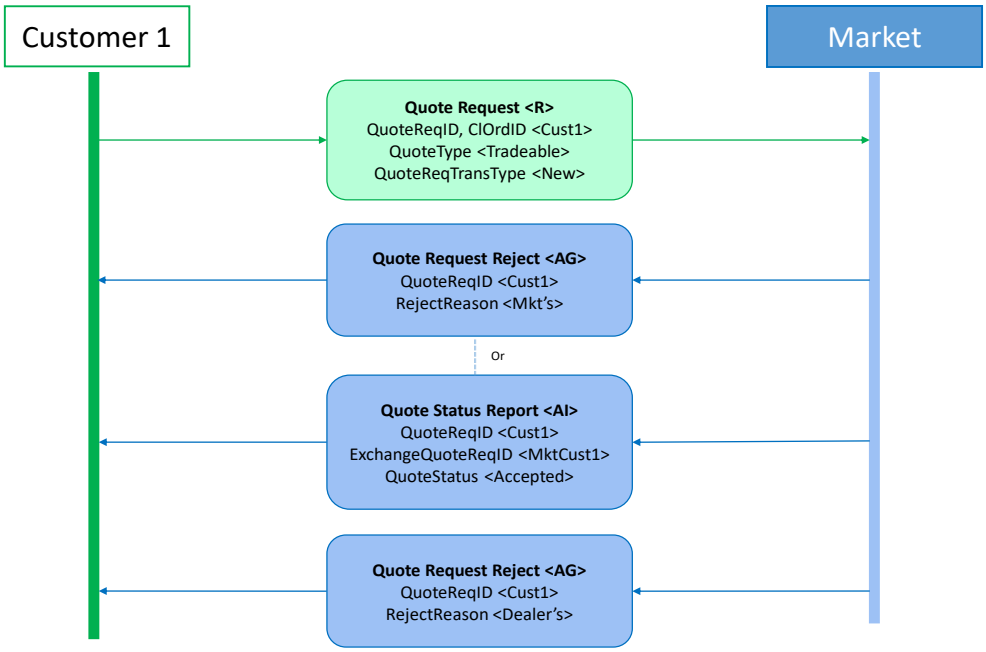
3.3.1 RFCQ workflow

The negotiation starts whenever the Customer (or buy-side) issues a Quote Request (R), identifying zero or more Dealer participants (or sell-side). If no participants are identified, the RFQ is intended to be forwarded to all the available Dealers.

The market (FTX platform) may respond with a Quote Request Reject (AG) if the validation fails. Otherwise, the request is acknowledged with a Quote Status report (AI) message and forwarded to all the intended Dealers. In the Quote Status Report (AI) message, the ExchangeQuoteReqID custom field (tag 9770) contains the unique ID assigned by the market to the RFQ.

If a Dealer rejects the request, the Customer will receive a Quote Request Reject (AG).

The flow is shown in the picture below.

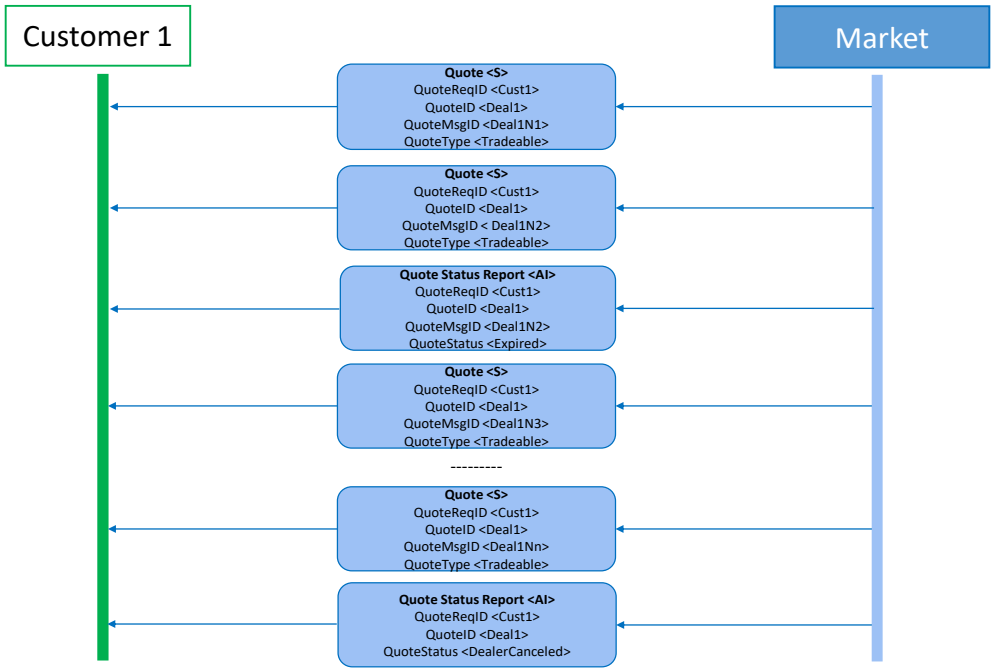


Any quote received by Dealers is forwarded to the Customer using the Quote message (S). The message contains a reference to the Quote Request it is responding to (QuoteReqID). A Quote (S) message contains three timestamps:

- *TransactTime* (tag 60): time of quote creation by the dealer
- *ValidUntilTime* (tag 62): Validity time of the quote for automatch. After the time set, if the quote is hit/lifted, the counterparty is asked for a last look (see below for the details).
- *ExpireTime* (tag 126): Expiration time of the quote. After the time set, the quote cannot be hit/lifted any longer.

A Dealer may edit a previously issued quote (streaming). The edited quote is forwarded to the Customer with a new Quote (S) message containing the same QuoteID (tag 117) and a new distinct QuoteMsgID (tag 1166). As a new quote edit is forwarded to the customer, the previous quote image with the same QuoteID is no longer valid and it cannot be hit/lifted any longer, even though the validity time and the expire time have not elapsed yet.

As soon as the ExpireTime of the current quote image elapses, a Quote Status Report (AI) message is forwarded to the Customer with QuoteStatus = Expired (7). Furthermore, a Dealer may cancel a quote at any time. In such case, a Quote Status Report (AI) message is forwarded to the Customer with QuoteStatus = Dealer Canceled (703).

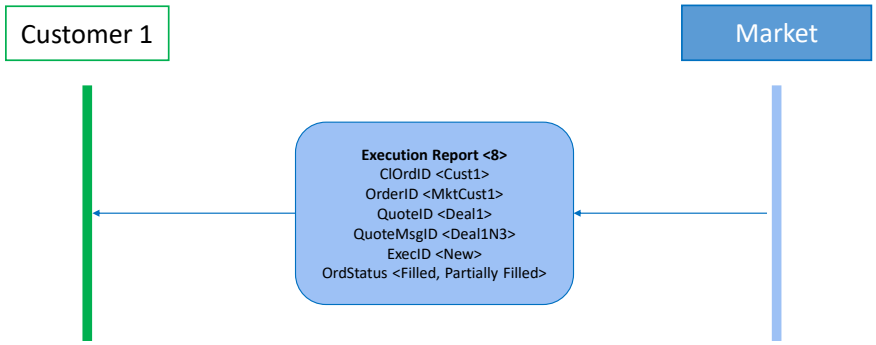


If the original quote request specifies a price level and a Dealer responds with a quote that specifies:

- same price as the request or better
- same quantity as the request or lower and in compliance with the minimum quantity

a deal is automatically generated and the customer receives an Execution Report message (8). If the requested quantity is not completely filled, depending on the market model, the request may remain active (in the Execution Report message: field 39 OrdStatus = 1 "Partially Filled") and may generate further deals with the same or other counterparties. The flow is shown in the picture below. The message contains the following identifiers:

- *ClOrderID*: contains the original unique identifier of the request (QuoteReqID)
- *OrderID*: contains the unique identifier associated by the market with the request, previously communicated in the ExchangeQuoteReqID tag.
- *QuoteID*: unique identifier of the quote that generated the deal
- *QuoteMsgID*: unique identifier of the quote image that generated the deal
- *ExecID*: unique identifier associated by the market with the deal.

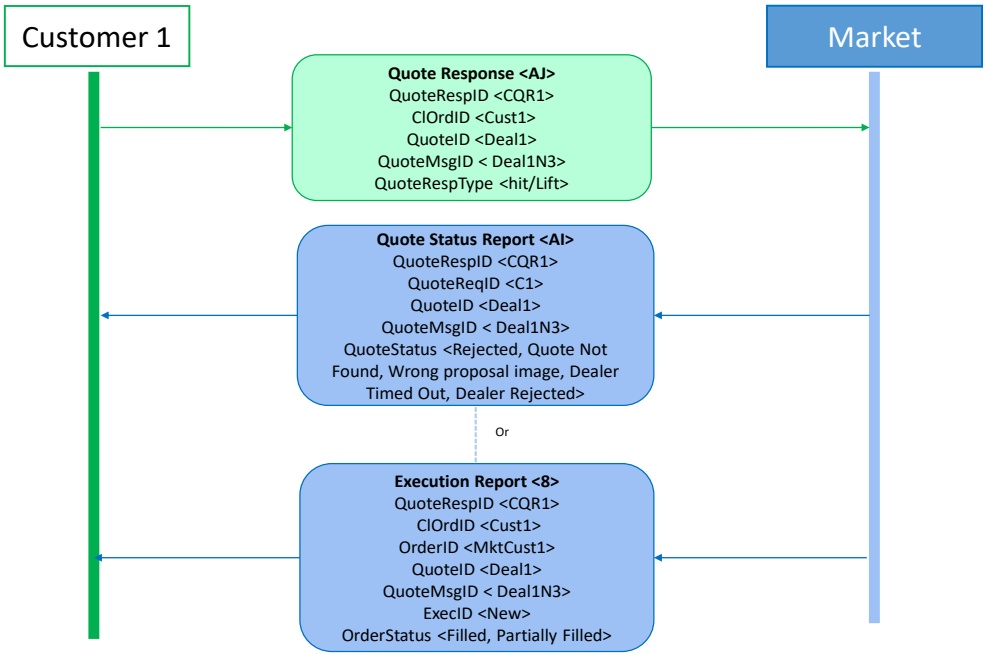


The Customer hits or lifts a Dealer's quote by sending a QuoteResponse (AJ) message to the market. The Quote Response must contain both the QuoteID identifier and the QuoteMsgID identifier to uniquely identify a specific Dealer's quote image. If the quote is hit/lifted after the validity time has expired, the Dealer is asked for a confirmation (last look). The Quote Response may be rejected with a QuoteStatusReport (AI) message for the following reasons:

Reason	Value	Quote Status
The QuoteResponse message failed validation (e.g. Quote Id does not exists, Quote Request timed out or is already completely filled, ...)	5	Rejected
The hit/lifted quote image is no longer valid because the dealer has edited the quote in the meantime (wrong proposal image)	700	Wrong proposal image
The hit/lifted quote does not exist or it is no longer valid because the dealer has cancelled the quote in the meantime, or the quote has been Hit/Lifted by a previous QuoteResponse.	9	Quote Not Found
The validity time of the proposal had expired, thus the Quote Request was sent to the dealer for last look. The dealer failed to answer in the allocated time.	701	Dealer Timed out
The validity time of the proposal had expired, thus the Quote Request was sent to the dealer for last look. The dealer rejected.	702	Dealer Rejected

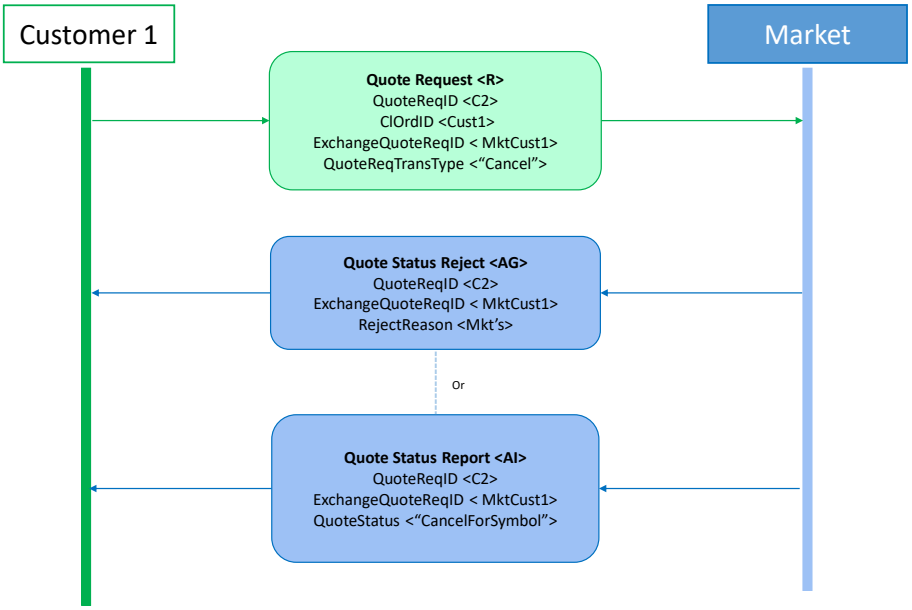
If the validity time has not expired or the Dealer confirm the last look, then a deal is generated and communicated to the customer via an Execution Report (8) message. If the requested quantity is not completely filled, depending on the market model the request may remain active (in the Execution Report message: field 39 OrdStatus = 1 "Partially Filled") and may generate further deals with the same or other counterparties.

The Quote Response flow described above is shown in the following picture.

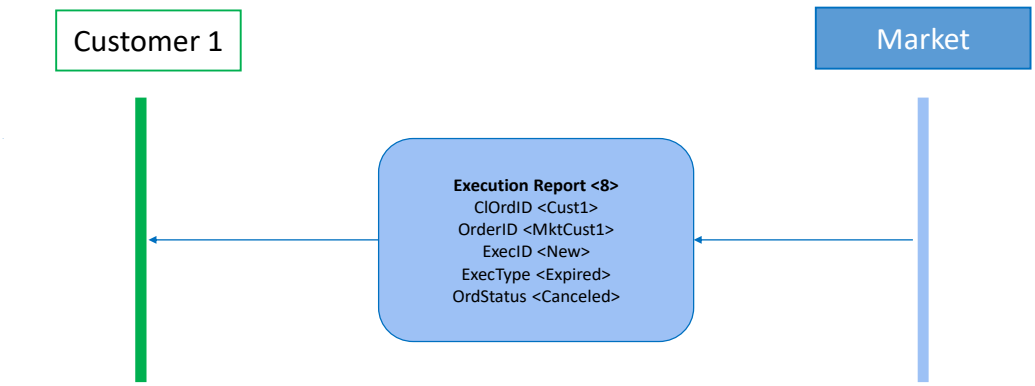


The QuoteResponse (AJ) message may be also used by the Customer to reject a quote and stop streaming from that specific Dealer (field 694 QuoteRespType = 6 "Pass").

The Customer can cancel an RFQ at any time by sending a Quote Request (R) message with field QuoteReqTransType (Tag 6530) with value C = Cancel, specifying the previously communicated ExchangeQuoteReqID related to the original Quote Request. The QuoteReqID field must be a new unique identifier for the message. If the cancellation fails (for example because the ExchangeQuoteReqID does not exist or it has already expired or it has already been completely filled), the market responds with a Quote Request Reject (AG) message, otherwise a Quote Status Report message is sent to acknowledge the cancellation. The flow is shown in the picture below.



When the timeout of an RFCQ which is still active expires, an Execution Report (8) message is sent to the customer with ExecType (150) = C (Expired) and OrdStatus(39) = 4 (Canceled).



In brief, a Customer always receives the following sequence of Execution Report (8) messages associated with a single Quote Request:

	field values	status of the request
zero or more messages	ExecType (150) = F (Trade) OrdStatus(39) = 1 (Partially filled)	The request remains active
one final message	ExecType (150) = F (Trade) OrdStatus(39) = 2 (Filled) or ExecType (150) = C (Expired) OrdStatus(39) = 4 (Canceled) or ExecType (150) = 4 (Canceled) OrdStatus(39) = 4 (Canceled)	The request is closed

The following sections detail the messages.

General note: For the fields with a "String" data type, a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared. The message will not be rejected.

Note that the side (tag 54) of transactions is always from the client point of view.

3.3.2 Quote Request (Type R)

Initiation of a Request for Quote by the client.

Note: For an Outright Quote Request, all trade parameters are described directly within the Repeating Group NoRelatedSym (Tag 146), while for a Switch Quote Request the following parameters are described within the sub repeating Group NoLegs (Tag 555) and are repeated once for each side of the trade:

- Security identification (SecurityID, SecurityIDSource)
- Side
- quantity-related parameters (OrderQty, MinQty)
- price-related parameters (Price, PriceType).

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = R		Y
131	QuoteReqID	Unique identifier assigned by the client to the message.	String (32)	Y
11	ClOrdID	Must be equal to QuoteReqID if QuoteReqTransType = New, otherwise it must be the same as in the original request.	String (32)	Y
100	ExDestination	Code of the market section.	String (24)	Y
6530	QuoteReqTransType	Identifies message transaction type. Valid values: • N = New (Default) • C = Cancel "Cancel" value is used to cancel a previously issued RFCQ. In this case, field ExchangeQuoteReqID (9770) is mandatory, while all other fields but QuoteReqID (131) and ClOrdID (11) are ignored.	Char	N
9770	ExchangeQuoteReqID	Mandatory if QuoteReqTransType (6530) = C. It must be filled with the unique identifier assigned by the market to the original request to be cancelled.	String (32)	C
58	Text	User text	String (256)	N
528	OrderCapacity	Valid values: • G = Proprietary • I = Individual (Client)	Char	Y
146	NoRelatedSym	Valid values: • 1	Integer	Y
→ 55	Symbol	Valid values: • '[N/A]'	String (12)	Y
→ 537	QuoteType	Valid values: • 1 = Tradable (default)	Char	N

→	63	SettlType	Indicates the settlement period. If present, SettlDate (64) overrides this field. If both SettlType (63) and SettDate (64) are omitted, the default value for the SettlType (63) is 0 (Regular). Regular is defined as the default settlement period for the particular security on the Exchange of Execution. Valid values: • 0 = Regular (default) • 1 = Cash • 2 = Next Day (T+1) • 3 = T+2 • 4 = T+3 • 5 = T+4 • 9 = T+ 5	Char	N
→	64	SettlDate	Settlement date YYYYMMDD	String (8)	N
→	1	Account	Account for which the contracts are to be bought or sold	String (32)	N
→	40	OrdType	Valid values: • 1 = Market (Auction) (default) • 2 = Limit (price limit)	Char	N
→	59	Timeinforce	Specifies whether the request has a timeout or is valid until the market closes. Valid values: • 0 = Day (default) • A = Good For Time (GFT) If value is 0 (Day), the ExposureDuration (1629) must be absent. The request is valid until the market closes. If value is A (GFT), the request expires after ExposureDuration (1629) seconds if present, otherwise after the default timeout set by the exchange.	Char	N
→	1629	ExposureDuration	Timeout duration in seconds. If not present and Timeinforce (59) = GFT (A), the RFQ expires after a default timeout set by the exchange.	Integer	N
→	60	TransactTime	Time this request was initiated/released by the client. TimeStamp with following format: YYYYMMDD-HH:MM:SS.μμμμμμ, e.g. 20170620-10:06:51.453765	String (24)	Y
→	453	NoPartyIDs	This repeating group identifies the dealers or target lists to whom the request must be issued. For each party, all PartyID (448), PartyIDSource (447), and PartyRole (452) entries must be repeated. If missing, the request is issued to all dealers.	Integer	N

→	→	448	PartyID	Party identifier/code: Member code or Target List code assigned by the Exchange must be used. A Target List code identifies a predefined list of Members. The PartyID can contain a generic string only if PartyRole = 3 (ClientID). In this case PartyIDSource must be D (Custom Code). For other values of PartyIDSource, PartyID must contain a number (number 0,1,2,3 are currently reserved for NoClient,ALGO,PNAL and CLIENT) Note that if PartyRole = 3 or 5, the Party Identifier must be between 0 and 4294967295	String (32)	Y
→	→	447	PartyIDSource	Identifies class or source of the PartyID (448) value. Valid value: • D = Proprietary/Custom code • M = Algorithm Short Code • N = Natural Person Short Code	Char	Y
→	→	452	PartyRole	Identifies the type or role of the PartyID. Valid Value: • 3 = ClientID • 5 = Investor ID (the Investment Decision Maker) • 12 = Executing Trader (the Executing Decision Maker) • 700 = Target List	Char	Y
→	The following fields must be specified for an outright request. They will be ignored for multi-leg requests (field NoLegs specified with value >1)					
→	48	SecurityID		Security identifier value of SecurityIDSource (22) type, valid value ISIN. Requires SecurityIDSource.	String (12)	C
→	22	SecurityIDSource		Valid values: • 4 = ISIN number	Char	C
→	54	Side		Valid values: • 1 = Buy • 2 = Sell	Char	C
→	38	OrderQty		Quantity of the instrument to be traded	Integer	C
→	110	MinQty		Minimum quantity to be executed. Must be equal to or lower than OrderQty (38).	Integer	N
→	The following fields are managed only if NoLegs is not present (Outright request) or if NoLegs has value=2 (Switch request). They are ignored otherwise.					
→	423	PriceType		Code to represent the type of Price (44). For Outright request, the valid values are: • 1 = Percentage (default) • 9 = Yield For Switch request, the valid values are: • 6 = Spread (default)	Char	N
→	44	Price		For Outright request: Specifies the price limit for the request. Required if OrdType (40) = Limit, ignored otherwise. For Switch request: Specifies the requested Spread price for the two legs.	Float	C
→	The following fields must be specified for a multi-leg request.					

→	555		NoLegs	Valid value: • 2 The fields below must be repeated for each side of the trade	Integer	N
→	→	600	LegSymbol	Valid values: • '[N/A]'	String (12)	C
→	→	602	LegSecurityID	Security identifier value of LegSecurityIDSource (603) type, valid value ISIN. Requires LegSecurityIDSource.	String (12)	C
→	→	603	LegSecurityIDSource	Valid values: • 4 = ISIN number	Char	C
→	→	624	LegSide	Valid values: • 1 = Buy • 2 = Sell	Char	C
→	→	687	LegQty	Quantity of the instrument to be traded	Integer	C
→	→	7252	LegMinBidSize	The minimum bid amount for this leg. Zero is an admitted value	Integer	N
→	→	7254	LegMinOfferSize	The minimum offer amount for this leg. Zero is an admitted value	Integer	N
→	→	566	LegPrice	Specifies the leg price limit for the outright or Switch request. It is required if OrdType (40) = Limit and PriceType is not equal to Spread, otherwise it is ignored.	Float	C
	<Standard Trailer>					Y

3.3.3 Quote Response (Type AJ)

Used to hit/Lift or reject a Quote by the customer.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = AJ		Y
693	QuoteRespID	Unique identifier for Quote Response message	String (32)	Y
117	QuoteID	Unique identifier for the quote	String (32)	Y
1166	QuoteMsgID	Unique identifier for the quote message	String (40)	Y
11	ClOrdID	Required only when QuoteRespType = 1 (Hit/Lift). Must be equal to QuoteReqID of the original request.	String (32)	C
9770	ExchangeQuoteReqID	Required only when QuoteMsgID = MarketBest. Filled with the unique identifier assigned by the market to the original request.	String (32)	C

694	QuoteRespType	Valid values: • 1 = Hit/Lift • 6 = Pass	Char	Y
100	ExDestination	Code of the market section.	String (24)	Y
60	TransactTime	TimeStamp with following format: YYYYMMDD-HH:MM:SS.μμμμμμ, e.g. 20170620-10:06:51.453765	String (24)	Y
58	Text	User text	String (64)	N
55	Symbol	=[N/A]	String (12)	Y
The following fields: ○ are ignored when NoLegs > 1 ○ may be specified for a quote response related to an outright quote. ○ If not present, the response is intended to accept qty/price parameters of the quote image that is hit/lifted ○ If present, all of them must be specified				
48	SecurityID	Security identifier value of SecurityIDSource (22) type, valid value ISIN. Requires SecurityIDSource.	String (12)	N
22	SecurityIDSource	Valid values: • 4 = ISIN number	Char	N
The following field: ○ are ignored when NoLegs > 1 ○ are ignored when QuoteRespType = 6 (Pass) ○ may be specified for a quote response related to an outright quote. ○ If not present, the response is intended to accept qty/price parameters of the quote image that is hit/lifted ○ If present, all of them must be specified				
54	Side	Valid values: • 1 = Buy • 2 = Sell	Char	N
423	PriceType	Code to represent the type of BidPx/OfferPx. Valid values are: • 1 = Percentage (default) • 9 = Yield	Char	N
132	BidPx	Bid Price. Present only if Side = 1	Float	C
133	OfferPx	Offer Price. Present only if Side = 2	Float	C
134	BidSize	Bid quantity. Present only if Side = 1	Integer	C
135	OfferSize	Offer quantity. Present only if Side = 2	Integer	C

The following fields:
• are ignored when QuoteRespType = 6 (Pass)
• may be specified for a quote response related to a multileg quote.
  ○ If not present, the response is intended to accept qty/price parameters of the quote image that is hit/lifted
  ○ If present, all the fields below must be specified					

555		NoLegs	Valid value: 2 The fields below must be repeated once for each side of the trade	Integer	N
→	600	LegSymbol	Valid values: • '[N/A]'	String (12)	C
→	602	LegSecurityID	Security identifier value of LegSecurityIDSource (603) type, valid value ISIN. Requires LegSecurityIDSource.	String (12)	C
→	603	LegSecurityIDSource	Valid values: • 4 = ISIN number	Char	C
→	624	LegSide	Valid values: • 1 = Buy • 2 = Sell	Char	C
→	687	LegQty	Quantity of the instrument to be traded	Integer	C
→	686	LegPriceType	Code to represent the price type. Valid values: • 1 = Percentage (default) • 9 = Yield	Char	N
→	681	LegBidPx	Bid Price. Present only if LegSide = 1	Float	C
→	684	LegOfferPx	Offer Price. Present only if LegSide = 2	Float	C
		<Standard Trailer>			Y

3.3.4 Quote Request Reject (Type AG)

Used to reject a new RFCQ or a Cancel request that cannot be fulfilled. In case of a new RFCQ, the author of the rejection may be either the market or a counterparty: if QuoteRequestRejectReason = 10 (Pass) then the author of the rejection is a counterparty, otherwise it is the market.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = AG		Y
131	QuoteReqID	Identifier of the original Quote Request (R) message..	String (32)	Y
9770	ExchangeQuoteReqID	Present only if the original message was a cancellation request. Filled with the unique identifier assigned by the market to the original request to be canceled.	String (32)	C

658		QuoteRequestRejectReason		Valid values: • 1 = Unknown symbol (Security) • 2 = Exchange (Security) closed • 5 = Invalid price • 6 = Not authorized to request quote • 10 = Pass • 99 = Other If value is 99, further details are given in field Text (58). If value is 10, the originator of the rejection is a counterparty, otherwise it is a market rejection.	Char	N
58		Text		Contains details about the reason of the rejection in case QuoteRequestRejectReason = 99	String (64)	N
146		NoRelatedSym		Valid values: • 1	Integer	Y
→	55	Symbol		Valid values: • '[N/A]'	String (12)	Y
→	453	NoPartyIDs		Present only if the author of the rejection is a counterparty. Value: 1	Integer	C
→	→	448	PartyID	Party identifier/code. Member code assigned by the Exchange must be used.	String (32)	Y
→	→	447	PartyIDSource	Identifies class or source of the PartyID (448) value. Valid value: • D = Proprietary/Custom code • M = Algorithm Short Code • N = Natural Person Short Code	Char	Y
→	→	452	PartyRole	Identifies the type or role of the PartyID. Valid Value: • 3 = ClientID • 5 = Investor ID (the Investment Decision Maker) • 12 = Executing Trader (the Executing Decision Maker)	Char	Y
		<Standard Trailer>			Y	

3.3.5 Quote (Type S)

Used to convey Dealer's quotes.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = S		Y
131	QuoteReqID	Unique identifier for Quote Request (R) message the quote is a response to.	String (32)	Y
117	QuoteID	Unique identifier of the quote	String (32)	Y

1166		QuoteMsgID	Unique identifier of the quote message	String (40)	Y
537		QuoteType	Valid values: • 1 = Tradable (default)	Char	Y
100		ExDestination	Code of the market section.	String (24)	Y
453		NoPartyIDs	The number of parties specified in the repeating group. Valid values: • 1	Integer	Y
→	448	PartyID	Identifier/code of the party that provided the quote. The Member code assigned by the Exchange must be used.	String (32)	Y
→	447	PartyIDSource	Identifies class or source of the PartyID (448) value. Valid values: • D = Proprietary/Custom code • M = Algorithm Short Code • N = Natural Person Short Code	Char	Y
→	452	PartyRole	Identifies the type or role of the PartyID. Valid values: • 1 = Executing Firm	Char	Y
62		ValidUntilTime	Validity time of the quote for automatch. After this time, if the quote is hit/lifted, the counterparty is asked for last look. TimeStamp with following format: YYYYMMDD-HH:MM:SS.μμμμμμ, e.g 20170620-10:06:51.453765	String (24)	N
126		ExpireTime	E.g.piration time of the quote, after which the quote cannot be hit/lifted any longer. TimeStamp with following format: YYYYMMDD-HH:MM:SS.μμμμμμ, e.g. 20170620-10:06:51.453765	String (24)	N
60		TransactTime	TimeStamp with following format: YYYYMMDD-HH:MM:SS.μμμμμμ, e.g. 20170620-10:06:51.453765	String (24)	Y
58		Text	User text	String (64)	N
55		Symbol	=[N/A]	String (12)	Y
The following fields must be specified for a quote related to an outright request. They will be ignored for multi-leg quotes (field NoLegs specified with value >1)					
48		SecurityID	Security identifier value of SecurityIDSource (22) type, valid value ISIN. Requires SecurityIDSource.	String (12)	C
22		SecurityIDSource	Valid values: • 4 = ISIN number	Char	C

54		Side	Valid values: • 1 = Buy • 2 = Sell	Char	C
423		PriceType	Code to represent the type of BidPx/OfferPx. Valid values are: • 1 = Percentage (default) • 9 = Yield	Char	N
132		BidPx	Bid Price. Present only if Side = 1	Float	C
133		OfferPx	Offer Price. Present only if Side = 2	Float	C
647		MinBidSize	Minimum bid quantity. Present only if Side = 1. Zero is an admitted value	Integer	C
134		BidSize	Bid quantity. Present only if Side = 1	Integer	C
648		MinOfferSize	Minimum offer quantity. Present only if Side = 2. Zero is an admitted value	Integer	C
135		OfferSize	Offer quantity. Present only if Side = 2	Integer	C
The following fields must be specified for a multi-leg request.					
555		NoLegs	Valid value: 2 The fields below must be repeated for each side of the trade	Integer	N
→	602	LegSecurityID	Security identifier value of LegSecurityIDSource (603) type, valid value ISIN. Requires LegSecurityIDSource.	String (12)	C
→	603	LegSecurityIDSource	Valid values: • 4 = ISIN number	Char	C
→	624	LegSide	Valid values: • 1 = Buy • 2 = Sell	Char	C
→	687	LegQty	Quantity of the instrument to be traded	Integer	C
→	7252	LegMinBidSize	The minimum bid amount for this leg. Zero is an admitted value	Integer	N
→	7254	LegMinOfferSize	The minimum offer amount for this leg. Zero is an admitted value	Integer	N
→	686	LegPriceType	Code to represent the price type. Valid values: • 1 = Percentage (default) • 9 = Yield	Char	N
→	681	LegBidPx	Bid Price. Present only if LegSide = 1	Float	C
→	684	LegOfferPx	Offer Price. Present only if LegSide = 2	Float	C
		<Standard Trailer>			Y

3.3.6 Quote Status Report (AI)

Used either to acknowledge a Quote Request (new RFCQ or Cancellation Request) or to reject a Quote Response, or unsolicited to notify status changes of quotes.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = AI		Y
131	QuoteReqID	Identifier of the original Quote Request (R) message. Present only when responding to a Quote Request message or in unsolicited messages or when responding to a Quote Response with QuoteMsgID = MarketBest.	String (32)	C
9770	ExchangeQuoteReqID	Present only when responding to a Quote Request message or when responding to a Quote Response with QuoteMsgID = MarketBest. Contains the unique identifier assigned by the market to the original RFCQ.	String (32)	C
693	QuoteRespID	Identifier of the original Quote Response (AJ) message. Present only when responding to a Quote Response message.	String (32)	C
117	QuoteID	Unique identifier for the quote. Present only when responding to a Quote Response message, when QuoteMsgID is not equal to MarketBest, or in unsolicited messages.	String (32)	C
1166	QuoteMsgID	Unique identifier for the quote message. Present only when responding to a Quote Response message, when QuoteMsgID is not equal to MarketBest or in unsolicited messages.	String (40)	C
55	Symbol	• = [N/A]	String (12)	Y
297	QuoteStatus	Valid values when the message is positively acknowledging a QuoteRequest message to create a new RFCQ: • 0 = Accepted Valid values when the message is positively acknowledging a QuoteRequest message to cancel an existing RFQ : • 1 = Canceled for Symbol(s) Valid values for unsolicited messages: • 7 = Expired • 703 = Dealer Canceled • 704 = Cancelled by Governance Valid values when the message is used to reject a QuoteResponse message: • 5 = Rejected • 9 = Quote Not Found • 700 = Wrong proposal image • 701 = Dealer Timed out • 702 = Dealer Rejected If value is 5, further details are given in field Text (58).	char	Y
58	Text	Contains details about the reason of the rejection in case QuoteStatus = 5	String (64)	N
	<Standard Trailer>			Y

The following schema summarize how to use identifiers and QuoteStatus values in the various scenarios.

Identifiers	Positive acknowledge of a Quote Request	Rejection of a Quote Response	Unsolicited
QuoteReqID (131)	Y	Y	Y
ExchangeQuoteReqID (9770)	Y	--	--
QuoteRespID (693)	--	Y	--
QuoteID (117)	--	Y	Y
QuoteMsgID (1166)	--	Y	Y
QuoteStatus			
valid values	• 0 = accepted • 1 = cancel for symbol	• 5 = Rejected • 9 = Quote Not Found • 700 = Wrong proposal image • 701 = Dealer Timed out • 702 = Dealer Rejected	• 7 = Expired • 703 = Dealer Canceled • 704 = Canceled by Gov

4 MARKET REFERENCE DATA APPLICATION MESSAGES

This chapter details the messages used by the client to request/receive market static data to/from the FTX Platform. It also describes the messages that FTX Platform sends automatically.

The following messages are currently supported:

Type	Name	Direction	Description
c	Security Definition Request	In	Request for security information
e	Security Status Request	In	Request for the status of a security
d	Security Definition	Out	Response to a Security Definition Request
f	Security Status	Out	Response to a Security Status Request
V	Market Data Request	In	Request to get the latest snapshot of all the market prices for a security
W	Market Data Snapshot / Full Refresh	Out	Reponses to a Market Data Request and automatic refresh in case data are modified in the FTX platform

4.1 SECURITY DEFINITION REQUEST (TYPE C)

This message enables the client to request information about a security to FTX Platform. Every valid Security Definition Request message would be responded with a Security Definition (type d) see section 4.4.

For the fields with a "String" data type, a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared. The message will not be rejected.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = c		Y
320	SecurityReqID	Unique ID of a Security Definition Request assigned by the client	String (63)	Y
321	SecurityRequestType	Type of Security Definition Request. Valid values: • 3 – request list securities	Integer	Y
	<Standard Trailer>			Y

4.2 SECURITY STATUS REQUEST (TYPE E)

The Security Status Request message enables the client to request the status of a security. Every valid Security Status Request message would be responded with a Security Status (type f) see section 0.

For the fields with a "String" data type, a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared. The message will not be rejected.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = e		Y
324	SecurityStatusReqID	ID of a Security Status Request assigned by the client	String (63)	Y
48	SecurityID	Exchange Security Identifier. Valid values can be retrieved by Security Definition Request enquiry.	String (64)	Y
22	SecurityIDSource	Valid value: • 8 = Exchange Symbol	Char	Y

263	SubscriptionRequestType	Subscription Request Type Valid values: • 0 = Snapshot • 1 = Snapshot + Updates (Subscribe) • 2 = Disable previous Snapshot + Update Request (Unsubscribe)	Char	Y
	<Standard Trailer>			Y

4.3 MARKET DATA REQUEST (TYPE V)

When the client logs on to FTX Platform, it can send a Market Data Request message to get the latest snapshot of all the market prices for the security. The response message for this request will be the Market data snapshot with the latest price, see section0. It will not send all the executed trades from the last logon.

For the fields with a "String" data type, a maximum length is specified. If a received string is longer than the maximum specified, it will be truncated to the maximum length declared it will be truncated to the maximum length declared. The message will not be rejected.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = V		Y
262	MDReqID	Request ID set by the client	String (63)	Y
263	SubscriptionRequestType	Subscription Request Type Valid value: • 1 = Snapshot + Updates (Subscribe)	Char	Y
264	MarketDepth	Depth of market for Book Snapshot Valid value: • 0 = Full Book (5 Level)	Integer	Y
265	MDUpdateType	Specifies the type of Market Data update. Valid value: • 0 = Full Refresh	Integer	Y
267	NoMDEntryTypes	Will be set to 0: All the available market entry types will be sent	Integer	N
146	NoRelatedSym	Number of symbols (i.e. Security) requested. Value always set to 0 – It means that Market data response will be sent for all securities.	Integer	N
4001	BookType	Indicates whether the client needs to subscribe the Depth (aggregated book), the Book or both. The possible values are: • P = Price book • O = Order Book • B = Both (default P).	Char	N

	<Standard Trailer>			Y

4.4 SECURITY DEFINITION (TYPE D)

In response to a Security Definition Request, the FTX Platform will send the complete list of all securities available for trading into the system. The FTX Platform will respond with one message for each security.

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = d		Y
320	SecurityReqID	Unique ID of a Security Definition Request assigned by the client	String (63)	Y
322	SecurityResponseID	Identifier for the Security Definition message	String (63)	Y
323	SecurityResponseType	Type of Security Definition message response. Valid values: • = 4 if list of securities is returned • = 5 if the request is rejected.	Integer	N
55	Symbol	ISIN Code	String (12)	Y
48	SecurityID	Exchange Security Identifier	String (64)	Y
22	SecurityIDSource	Valid value: • 8 = Exchange Symbol	Char	Y
460	Product	Valid values: • 3 = CORPORATE • 5 = EQUITY • 6 = GOVERNMENT • 12 = OTHER	Integer	Y
231	ContractMultiplier	Lot Size (minimum lot value of the security)	Float	Y
223	CouponRate	Coupon Rate	Float	N
106	Issuer	Name of security issuer	String (64)	N
107	SecurityDesc	Security Description	String (64)	N
15	Currency	Contain the standard international currency code, which represents the trading currency of the Security	String (3)	N

58	Text		Settlement information	String (6)	N
562	MinTradeVol		Minimum fill quantity	Qty	Y
5564	QtyTick		Qty tick	Float	Y
5565	TradingStartDate		Trading Start Date Format YYYYMMDD	String (8)	Y
5566	TradingStopDate		Trading Stop Date Format YYYYMMDD	String (8)	Y
541	MaturityDate		Maturity Date Format YYYYMMDD	String (8)	Y
5568	MinOrderQty		Minimum Order Qty	Qty	Y
336	TradingSessionID		Section of the security	String (24)	Y
5600	NoTicks		Number of tick values	Integer	Y
→	5601	MinPrice	Minimum price value for current tick	Float	Y
→	5602	MaxPrice	Maximum price value for current tick (0 means not specified)	Float	Y
→	5563	PriceTick	Tick value for price interval defined by MinPrice, MaxPrice	Float	Y
64	SettlDate		Settlement Date in YYYYMMDD format	Date	Y
461	CFICode		CFICode of the instrument	String (6)	Y
454	NoSecurityAltID		The number of parties specified in the repeating group. Always 1	Integer	Y
→	455	SecurityAltID	FISNCode	String (35)	Y
→	456	SecurityAltIDSource	FISNCode Source, value = W, this value could change when FIX protocol committee will decide about it	Char	Y
4000	RFEEnabled		Specifies whether the liquidity provider can quote with the RFE indicator set to subject or not. Possible values: • 0 = No • 1 = Yes For bonds and equities RFE is not applicable, therefore this tag is always set equal to 0	Char	N
311	UnderlyingSymbol		Undelying security's Symbol	String (12)	N
	<Standard Trailer>				Y

4.5 SECURITY STATUS (TYPE F)

The FTX Platform uses this message as an answer to a security status request (type e).

Tag	Field Name	Content	Data Type	Req
	<Standard Header>	MsgType <35> = f		Y
324	SecurityStatusReqID	Request ID set by the client	String (63)	Y
48	SecurityID	Exchange Security Identifier	String (64)	Y
22	SecurityIDSource	Valid value: • 8 = Exchange Symbol	Char	Y
55	Symbol	ISIN code	String (12)	Y
326	SecurityTradingStatus	Identifies the trading status applicable: it has to be defined according to FTX Platform. Valid values: • 17 = Ready to trade (start of session) • 2 = Trading Halt • 18 = Not Available for trading (end of session)	Integer	Y
336	TradingSessionID	Identifies the phase of the market the security belongs to Valid values: • "CLO" = Closure • "PRA" = Pre Auction • "AUC" = Auction • "PRT" = Pre Trading • "NEG" = Negotiation • "NOP" = No Operation • "CRB" = Circuit Breaker • "MLQ" = Missing Liquidity (Provider) • "MGT" = Management	String (3)	Y
	<Standard Trailer>			Y

4.6 MARKET DATA - SNAPSHOT /FULL REFRESH (TYPE W)

The client will not have a mechanism to request all the market data messages sent since the last disconnection. The client can request a complete market snapshot on re-connection to have the latest market data.

Market data snapshot full refresh is used in two different ways:

1 it is the answer to a Market Data Request, see Market Data Snapshot full Refresh (type W) used as a response

2 It is an automatic refresh sent if a modification occurs in FTX Platform. The modifications can generate the 3 types of messages (Market Data Snapshot full refresh) described below:

- Static market data: This message, which provides market open and closing prices, sends opening or closing price changes, see Market Data Snapshot full Refresh (type W) used as a response **(MDEntryType = 4 and 5)**
- A Market Data message is sent on every change to the order book and represents the top 5 price levels of the book, see Market Data - Snapshot / Full Refresh Sent when there is a change in the Order Book **(MDEntryType = 0 or 1)**
- When a trade occurs, it will also result in a Market Data message being sent, see Market Data - Snapshot / Full Refresh Sent following every Trade **(MDEntryType = 2)**

4.6.1 Market Data Snapshot full Refresh (type W) used as a response

This message will be sent in response to a Market Data request. There will be one message for every instrument in the market. The client must send a request only once during a session. If the client application re-connects, it can send a request again.

Tag		Field Name	Content	Data Type	Req
		<Standard Header>	MsgType <35> = W		Y
262		MDReqID	Request ID set by the client	String (63)	N
48		SecurityID	Exchange Security Identifier	String (64)	N
22		SecurityIDSource	Valid value: • 8 = Exchange Symbol	String (1)	Y
55		Symbol	ISIN Code	String (12)	Y
268		NoMDEntries	Number of entries	Integer	Y
→	269	MDEntryType	Must be the first field in this repeating group. Type Market Data entry. Possible value: • 2 = Trade	Char	Y

→	270	MDEntryPx	Last Traded Price. (Nine integer places and Five decimal places.)E.g.. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Last Traded Quantity	Float	N
→	272	MDEntryDate	Trade Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Trade Time HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	274	TickDirection	Direction of the "tick". Valid values: • 0 – Plus Tick • 1 – Zero tick • 2 – Minus Tick	Char	N
→	1024	MDOriginType	It maps MMT Level 1 "Market Mechanism". Possible values are: • 0 = Book • 3 = Quote driven market • 4 = Dark order book • 1 = Off-book • 5 = Auction driven market • 6 = Quote negotiation • 8 = Hybrid market The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	5463	TradeID	Unique identifier of the trade that generated	String	N
→	625	TradingSessionSubID	This tag, together with tag 574, maps the MMT level 2: "Trading Mode". Possible values are: • 8 = Any auction • 2 = Opening or opening auction • 4 = Closing or closing auction • 6 = Intraday auction • 9 = Unscheduled intraday auction • 3 = (Continuous) trading • 5 = Post-trading • 10 = Out of main session trading The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N

→	574	MatchType	This tag, together with tag 625, maps the MMT Level 2 "Trading Mode". Possible values are: • 3 = Confirmed trade report (reporting from recognized markets) • 1 = One Party Trade Report (privately negotiated trade) • 9 = Systematic Internalizer The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N
→	828	TrdType	This tag, together with tag 1839, maps the MMT Level 3.1 "Transaction Type, Transaction Category". Possible values are: • 62 = Dark trade • 65 = Package trade • 2 = Exchange for physical The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	829	TrdSubType	This tag maps the MMT level 3.3: "Transaction Type: Agency Cross Indicator". Possible values are: • 37 = Crossed trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1838	NoTradePriceConditions	Number of TradePriceCondition (1839) entries	Integer	N
→→	1839	TradePriceCondition	This tag, together with tag 828, maps the MMT Level 3.1 "Transaction Type, Transaction Category", MMT Level 3.6 "Special Dividend", MMT Level 3.8 "Transaction Type: Ordinary Trades or Trades outside price formation". Possible values are: • 13= Special Dividend • 14 = Price improvement • 15 = Non-price forming trade • 16 = Trade exempt from trading obligation • 17 = Price is pending The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1115	OrderCategory	This tag, together with tags 2669 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". Possible values are • 3 = Privately negotiated trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Char	N

→	2668	NoTrdRegPublications	Number of TrdRegPublicationType (2669) and TrdRegPublicationReason (2670) entries.	Integer	N
→→	2669	TrdRegPublicationType	This tag, together with tags 1115 and tag 2669, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are • 0 = Pre-trade transparency waiver • 1 = Post-trade deferral The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→→	2670	TrdRegPublicationReason	This tag, together with tags 1115 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are: • 1 = No preceding order in book as transaction price depends on system-set reference price for an illiquid instrument) • 0 = No preceding order in book as transaction price set within average spread of a liquid instrument) • 2 = No preceding order in book as transaction price is subject to conditions other than current market price • 3 = No public price for preceding order as public reference price was used for matching orders • 4 = No public price quoted as instrument is illiquid • 5 = No public price quoted as order size is above standard market size The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	277	TradeCondition	This tag, together with tags 2669 and 2670, maps MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator". Possible values are • 6= Benchmark The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N

→	2405	ExecMethod	This tag maps Level 3.7 "Transaction Type: Off Book Automated Indicator". Possible values are • 0 = Undefined/unspecified • 1 = Manual (the transaction was executed in a manual or other non-automated manner) • 2 = Automated (the transaction was executed on an automated execution platform such as an automated systematic internalizer system, broker crossing network, dark pool trading, "direct to capital" systems, broker position unwind mechanisms, etc.) The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	2667	AlgorithmicTradeIndicator	This tag, together with tags 2669 and tag 2670, maps MMT Level 3.9 "Transaction Type: algorithmic indicator". Possible values are • 1 = Algorithmic trade • 0 = Non-algorithmic trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1390	TradePublishIndicator	This tag maps MMT Level 4.1 "Publication Mode and Post Trade Deferral". Possible values are • 1 = Publish trade • 2 = Deferred publication The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1934	RegulatoryReportType	This tag maps MMT Level 4.2 "Publication Mode and Post Trade Deferral". Possible values are: • 11 = Limited details trade • 12 = Daily aggregated trade • 13 = Volume omission trade • 14 = Four weeks aggregation trade • 15 = Indefinite aggregation trade • 16 = Volume omission trade, eligible for subsequent enrichment in aggregated form • 17 = Full details of earlier "limited details trade" • 18 = Full details of earlier "daily aggregated trade" • 19 = Full details of earlier "volume omission trade" • 20 = Full details of earlier "four weeks aggregation trade" • 21 = Full details of earlier "volume omission trade, eligible for subsequent enrichment in aggregated form" The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N

→	269	MDEntryType	Possible value: • 4 = Theoretical Auction Price	Char	Y
→	270	MDEntryPx	Theoretical Auction Price. (Nine integer places and Five decimal places.)E.g.. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Quantity traded at the Theoretical Auction Price	Float	N
→	272	MDEntryDate	Trade Date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • 5 = Closing Price	Char	Y
→	270	MDEntryPx	Closing Price (previous day closing price). (Nine integer places and Five decimal places.)E.g.. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Total volume (quantity) traded.	Float	Y
→	272	MDEntryDate	Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • Q = Auction Closing Price, which is the weighted average price calculated on the basis of executed contracts during the Auction phase. (Note: this value is not available for all instruments and is not standard in version FIX 4.4)	Char	Y
→	270	MDEntryPx	Auction Closing Price (Nine integer places and Five decimal places.). E.g.. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Quantity traded at the Auction Closing Price	Float	N
→	272	MDEntryDate	Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • 7 = High Price	Char	Y
→	270	MDEntryPx	Highest Price for the day. (Nine integer places and Five decimal places.). E.g.. 999999999.99999. For more details, see page 73	Float	Y

→	272	MDEntryDate	Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • 8 = Lowest Price	Char	Y
→	270	MDEntryPx	Lowest Price for the day (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	272	MDEntryDate	Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • 0 = Bid	Char	Y
→	270	MDEntryPx	Bid Price level. (Nine integer places and Five decimal places.). E.g.. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Best Bid Quantity	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	346	NumberOfOrders	Number of orders at this price level	Integer	N
→	269	MDEntryType	Possible value: • 1 = Offer	Char	Y
→	270	MDEntryPx	Offer Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Best offer quantity	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	346	NumberOfOrders	Number of orders at this price level	Integer	N

→	269	MDEntryType	Possible value: • B = Trade Volume	Char	Y
→	270	MDEntryPx	Average Price (Nine integer places and Five decimal places.). E.g.. 999999999.99999. For more details, see page 73d	Float	Y
→	271	MDEntrySize	Total volume (quantity) traded.	Float	Y
→	269	MDEntryType	Possible value: • b = Order bid	Char	Y
→	270	MDEntryPx	Bid Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Bid quantity	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	37	OrderID	Contains the order ID or the quote ID. This field is mandatory if tag 269 is set to "b".	String (16)	C
→	290	MDEntryPositionNo	Number that indicates the quote position in the book.	Float	C
→	4002	IsTradable	Possible values: • A = Automatic • M= Manual	Char	Y
→	453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries. Possible parties entries: • 1-First Party: is the member • 2-Second Party: is the member operator	Integer	N
→	->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452).	String (16)	N
→	->447	PartyIDSource	Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Note: applicable values depend upon PartyRole (452) specified. • D = Proprietary/Custom code	Char	N
→	->452	PartyRole	Identifies the type or role of the PartyID (448) specified. Valid Value • 1 = Executing Firm • 12 = Executing Trader (associated with Executing Firm - actually executes)	Integer	N

→	269	MDEntryType	Possible value: • o = Order Offer	Char	Y
→	270	MDEntryPx	Offer Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Offer quantity	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	37	OrderID	Contains the order ID or the quote ID. This field is mandatory if tag 269 is set to "o".	String (16)	C
→	290	MDEntryPositionNo	Number that indicates the quote position in the book.	Float	C
→	4002	IsTradable	Possible values: • A = Automatic • M= Manual	Char	Y
→	453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries. Possible parties entries: • 1-First Party: is the member • 2-Second Party: is the member operator	Integer	N
→	->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452).	String (16)	N
→	->447	PartyIDSource	Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Note: applicable values depend upon PartyRole (452) specified. • D = Proprietary/Custom code	Char	N
→	->452	PartyRole	Identifies the type or role of the PartyID (448) specified. Valid Value • 1 = Executing Firm • 12 = Executing Trader (associated with Executing Firm - actually executes)	Integer	N
		<Standard Trailer>			Y

4.6.2 Market Data - Snapshot / Full Refresh Sent following every Trade

Tag		Field Name	Content	Data Type	Req
		<Standard Header>	MsgType <35> = W		Y
262		MDReqID	Contains the string "dummyid", because it is sent by the system unsolicited	String (63)	N
48		SecurityID	Exchange Security Identifier	String (64)	N
22		SecurityIDSource	Valid value: • 8 = Exchange Symbol	String (1)	Y
55		Symbol	ISIN Code	String (12)	Y
268		NoMDEntries	Number of Entries	Integer	Y
→	269	MDEntryType	• 2 – Trade	Char	Y
→	1024	MDOriginType	It maps MMT Level 1 "Market Mechanism". Possible values are: • 0 = Book • 3 = Quote driven market • 4 = Dark order book • 1 = Off-book • 5 = Auction driven market • 6 = Quote negotiation • 8 = Hybrid market The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	5463	TradeID	Unique identifier of the trade that generated	String	N
→	625	TradingSessionSubID	This tag, together with tag 574, maps the MMT level 2: "Trading Mode". Possible values are: • 8 = Any auction • 2 = Opening or opening auction • 4 = Closing or closing auction • 6 = Intraday auction • 9 = Unscheduled intraday auction • 3 = (Continuous) trading • 5 = Post-trading • 10 = Out of main session trading The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N

→	574	MatchType	This tag, together with tag 625, maps the MMT Level 2 "Trading Mode". Possible values are: • 3 = Confirmed trade report (reporting from recognized markets) • 1 = One Party Trade Report (privately negotiated trade) • 9 = Systematic Internalizer The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N
→	828	TrdType	This tag, together with tag 1839, maps the MMT Level 3.1 "Transaction Type, Transaction Category". Possible values are: • 62 = Dark trade • 65 = Package trade • 2 = Exchange for physical The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	829	TrdSubType	This tag maps the MMT level 3.3: "Transaction Type: Agency Cross Indicator". Possible values are: • 37 = Crossed trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1838	NoTradePriceConditions	Number of TradePriceCondition (1839) entries	Integer	N
→→	1839	TradePriceCondition	This tag, together with tag 828, maps the MMT Level 3.1 "Transaction Type, Transaction Category", MMT Level 3.6 "Special Dividend", MMT Level 3.8 "Transaction Type: Ordinary Trades or Trades outside price formation". Possible values are: • 13= Special Dividend • 14 = Price improvement • 15 = Non-price forming trade • 16 = Trade exempt from trading obligation • 17 = Price is pending The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1115	OrderCategory	This tag, together with tags 2669 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". Possible values are • 3 = Privately negotiated trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Char	N

→	2668	NoTrdRegPublications	Number of TrdRegPublicationType (2669) and TrdRegPublicationReason (2670) entries.	Integer	N
→→	2669	TrdRegPublicationType	This tag, together with tags 1115 and tag 2669, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are • 0 = Pre-trade transparency waiver • 1 = Post-trade deferral The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→→	2670	TrdRegPublicationReason	This tag, together with tags 1115 and tag 2670, maps Level 3.2 "Transaction Type: Negotiation Indicator". It maps also MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator", MMT Level 3.9 "Transaction Type: algorithmic indicator" and MMT Level 4.1 "Publication Mode and Post Trade Deferral" Possible values are: • 1 = No preceding order in book as transaction price depends on system-set reference price for an illiquid instrument) • 0 = No preceding order in book as transaction price set within average spread of a liquid instrument) • 2 = No preceding order in book as transaction price is subject to conditions other than current market price • 3 = No public price for preceding order as public reference price was used for matching orders • 4 = No public price quoted as instrument is illiquid • 5 = No public price quoted as order size is above standard market size The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	277	TradeCondition	This tag, together with tags 2669 and 2670, maps MMT Level 3.5 "Transaction Type: Benchmark or reference price indicator". Possible values are • 6= Benchmark The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	String	N

→	2405	ExecMethod	This tag maps Level 3.7 "Transaction Type: Off Book Automated Indicator". Possible values are • 0 = Undefined/unspecified • 1 = Manual (the transaction was executed in a manual or other non-automated manner) • 2 = Automated (the transaction was executed on an automated execution platform such as an automated systematic internalizer system, broker crossing network, dark pool trading, "direct to capital" systems, broker position unwind mechanisms, etc.) The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	2667	AlgorithmicTradeIndicator	This tag, together with tags 2669 and tag 2670, maps MMT Level 3.9 "Transaction Type: algorithmic indicator". Possible values are • 1 = Algorithmic trade • 0 = Non-algorithmic trade The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1390	TradePublishIndicator	This tag maps MMT Level 4.1 "Publication Mode and Post Trade Deferral". Possible values are • 1 = Publish trade • 2 = Deferred publication The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N
→	1934	RegulatoryReportType	This tag maps MMT Level 4.2 "Publication Mode and Post Trade Deferral". Possible values are • 11 = Limited details trade • 12 = Daily aggregated trade • 13 = Volume omission trade • 14 = Four weeks aggregation trade • 15 = Indefinite aggregation trade • 16 = Volume omission trade, eligible for subsequent enrichment in aggregated form • 17 = Full details of earlier "limited details trade" • 18 = Full details of earlier "daily aggregated trade" • 19 = Full details of earlier "volume omission trade" • 20 = Full details of earlier "four weeks aggregation trade" • 21 = Full details of earlier "volume omission trade, eligible for subsequent enrichment in aggregated form" The mapping between MMT level values and tag values can be found in the MMT document attached at the end of this document	Integer	N

→	270	MDEntryPx	Last Traded Price. (Nine integer places and Five decimal places.). E.g.. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Last Traded Quantity	Float	N
→	272	MDEntryDate	Trade Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Trade Time HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	274	TickDirection	• 0 – Plus Tick • 1 – Zero Plus tick • 2 – Minus Tick	Char	N
→	269	MDEntryType	• 4 – Theoretical Auction Price	Char	Y
→	270	MDEntryPx	Theoretical Auction Price. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Quantity traded at the Theoretical Auction Price	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	269	MDEntryType	• 5 – Closing Price	Char	Y
→	270	MDEntryPx	Closing Price (previous closing price till the day closing price is determined). (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Total volume (quantity) traded.	Float	Y
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • Q = Auction Closing Price which is the weighted average price calculated on the basis of executed contracts during the Auction phase. (Note: this value is not available for all instruments and is not standard in version FIX 4.4)	Char	Y

→	270	MDEntryPx	Auction Closing Price (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Price	Y
→	271	MDEntrySize	Quantity traded at the Auction Closing Price	Float	N
→	272	MDEntryDate	Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	• 7 – High Price	Char	Y
→	270	MDEntryPx	High Price for the day. (Nine integer places and Five decimal places.) E.g.. 999999999.99999. For more details, see page 73	Float	Y
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	269	MDEntryType	• 8– Low Price	Char	Y
→	270	MDEntryPx	Low Price. (Nine integer places and Five decimal places.) E.g. 999999999.99999. For more details, see page 73	Float	Y
→	272	MDEntryDate	Closing Price Trading date YYYYMMDD	String (8)	N
→	269	MDEntryType	Possible value: • B = Trade Volume	Char	Y
→	270	MDEntryPx	Average Price (Nine integer places and Five decimal places.)E.g. 999999999.99999. For more details, see page 73d	Float	Y
→	271	MDEntrySize	Total volume (quantity) traded.	Float	Y
		<Standard Trailer>			Y

4.6.3 Market Data - Snapshot / Full Refresh Sent when there is a change in the Order Book

This message is sent by the FTX Platform whenever there is a change in the Order Book Price levels. The group will contain bids first and then offers. If only one side of the book is available, the group will contain only bid price levels or offer price levels. FTX Platform will send the best prices in bid and the best prices in ask. Depending

on the configuration (market/section/member), FTX Platform can send up to the first twenty price levels, in increments of five. The first twenty prices in bid are sorted in descending order; the first twenty prices in offer are sorted in ascending order. This message will only be sent if there is a price level change. The full snap shot of the price levels is always sent.

Tag		Field Name	Content	Data Type	Req
		<Standard Header>	MsgType <35> =W		Y
262		MDReqID	Contains the string "dummyid", because it is sent by the system unsolicited.	String (63)	N
48		SecurityID	Exchange Security Identifier	String (64)	N
22		SecurityIDSource	Valid value: • 8 = Exchange Symbol	String (1)	Y
55		Symbol	ISIN Code	String (12)	Y
268		NoMDEntries	Sum of the price levels of bids and offers currently defined in the market. The possible values are: 10, 20, 30, 40.	Integer	Y
→	269	MDEntryType	Possible values: • 0 = Bid • 1 = Offers	Char	Y
→	270	MDEntryPx	Bid or offer Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Quantity available in this bid or offer price level	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Update time HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	Y
→	346	NumberOfOrders	Number of orders at this price level	Integer	N
→	269	MDEntryType	Possible value: • b = Order bid	Char	Y
→	270	MDEntryPx	Bid Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Bid quantity	Float	N

→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	37	OrderID	Contains the order ID or the quote ID. This field is mandatory if tag 269 is set to "b".	String (16)	C
→	290	MDEntryPositionNo	Number that indicates the quote position in the book.	Float	C
→	4002	IsTradable	Possible values: • A = Automatic • M= Manual	Char	Y
→	453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries. Possible parties entries: • 1-First Party: is the member • 2-Second Party: is the member operator	Integer	N
→	->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452).	String (16)	N
→	->447	PartyIDSource	Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Note: applicable values depend upon PartyRole (452) specified. • D = Proprietary/Custom code	Char	N
→	->452	PartyRole	Identifies the type or role of the PartyID (448) specified. Valid Value • 1 = Executing Firm • 12 = Executing Trader (associated with Executing Firm - actually executes)	Integer	N
→	269	MDEntryType	Possible value: • o = Order Offer	Char	Y
→	270	MDEntryPx	Offer Price level. (Nine integer places and Five decimal places.). E.g. 999999999.99999. For more details, see page 73	Float	Y
→	271	MDEntrySize	Offer quantity	Float	N
→	272	MDEntryDate	Trading Date YYYYMMDD	String (8)	N
→	273	MDEntryTime	Time sent from the Exchange HH:MM:SS.μμμμμμ, e.g. 10:06:51.453765	String (15)	N
→	37	OrderID	Contains the order ID or the quote ID. This field is mandatory if tag 269 is set to "o".	String (16)	C

→	290	MDEntryPositionNo	Number that indicates the quote position in the book.	Float	C
→	4002	IsTradable	Possible values: • A = Automatic • M= Manual	Char	Y
→	453	NoPartyIDs	Number of PartyID (448), PartyIDSource (447), and PartyRole (452) entries. Possible parties entries: • 1-First Party: is the member • 2-Second Party: is the member operator	Integer	N
→	->448	PartyID	Party identifier/code. See PartyIDSource (447) and PartyRole (452).	String (16)	N
→	->447	PartyIDSource	Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Note: applicable values depend upon PartyRole (452) specified. • D = Proprietary/Custom code	Char	N
→	->452	PartyRole	Identifies the type or role of the PartyID (448) specified. Valid Value • 1 = Executing Firm • 12 = Executing Trader (associated with Executing Firm - actually executes)	Integer	N
		<Standard Trailer>			Y

Note: Number Representation

The floating-point representation depends on the number that needs to be represented.

1) If a number is not zero but near to zero (including between -0.0001 and 0.0001): the number is represented as a floating-point number (for example, 0.0000135) or is represented with the scientific notation (for example, 1.35 e-5). The system chooses the representation that requires the least number of characters.

2) If a number is zero, or higher than 0.0001, or smaller than -0.0001: the number always has a floating-point representation (for example 0.000135).

Mapping between MMT Flags and FIX tags

The document with the mapping between MMT Flags and FIX tags is available at the following path:https://www.fixtrading.org/mmt/

APPENDIX

This subsection details the possible errors for order rejections.

Tag 58 (Possible value)
Security Not Active
Invalid Verb
Invalid Request Member
Invalid Request Operator
Invalid Validity Date
Invalid Validity Time
Invalid Request Status
Section Not Active
Order Not Found
Member Not Active
Operator Not Active
Undefined Section Phase
Undefined Security Phase
Invalid Security
Invalid Order Type
Price Overlap
Maximum Order Exceeded
Invalid Relative Qty
Invalid ConfirmFlag (only IDP)
Invalid IssueOrderFlag (only IDP)
Invalid TimeInForce
Invalid Position (only IDP)
Invalid Origin (only IDP)
Invalid Security Phase
Invalid Section Phase
Invalid Order Status

Invalid Order Qty
Invalid DisclosedQty
Invalid MinQty
Invalid MinFillSize
Invalid Order Price
Invalid MemberID
Invalid Request
Invalid CreditLine
Invalid IndirectLimit
Invalid Qty Parameter
Invalid TimeInForce Parameter
Numeric Overflow
Bid Price has violated Last Price Threshold
Ask Price has violated Last Price Threshold
Bid Price has violated Close Price Threshold
Ask Price has violated Close Price Threshold
Order Already Exist
Best Price Threshold Violated
AT Send Error
Security Not Tradable
Order can not match
Best Price Missing
Tag xxx should have value = yyy
Tag xxx has a wrong value
Only 1 PartyID is supported
Invalid Profile
Member not profiled on Class
Operator not profiled on Security Class
Operator not profiled on Qty Parameter
Operator not profiled on Capacity-TimeInForce

Operator not profiled for editing
Operator can not see other Operator Info
Invalid Bid MinVolQty
Invalid Ask MinVolQty
New Password Repeated
Invalid New Password Characters
New Password Too Easy
Insufficient New Password Length
Maximum Rfq exceeded
Maximum Rfq Quote exceeded
Invalid RFQ Quantity
Invalid RFQ Price
Invalid RFQ Min Qty
Invalid RFQ Destination Number
RFQ Reference Not Found
RFQ Type Not Allowed
RFQ Quote Not Found
RFQ Edit Not Allowed
Invalid RFQ Edit
RFQ Not Found
Not Administrator
Not Enabled Member TradeOn
ForwardForward Label Not Found
Calendar Not Found
Section Property Not Found
Invalid Section Property Value
Invalid Broken Date
Not Enabled CAPS Admin
Min Member Qty Not Enabled"
RFQ Not Allowed

RFQ Invalid Operator Role
Invalid RFQ Destination
Quote not active
Member Min Qty Not Enabled
Member Min Qty Not Compliant With Min Prod Quantity
Invalid RFQ Status
RFQ Not Active
Cannot create administrator
Invalid Hidden Qty
Order not active
Cannot change administrator
Trader Connected
Not Allowed to send Transaction
Counterpart Setup Error
Swap Welcome ValueDenied
Invalid Counterpart
Invalid Currency
Too Many Setup Switches
Invalid Section
Invalid Server Status
Invalid Equal Rate
Invalid Preview Initial Date
Invalid Preview Final Date
Invalid Member Status
Insufficient Guarantee
Fill Already Allocated
Invalid Allocation
Too many securities with same Issue Type
Too many securities with same Issue Country
Too many securities with same Issuer

Too many securities with same Rating
Too many securities with same Currency
Security Price too old for allocation
Security quantity too low for allocation
Cannot use securities issued by you
Invalid Initial Date
Invalid Final Date
Quote Not Tradable
Invalid Settlement Mode
Invalid Transparent Flag
Invalid CCP Only Flag
Invalid Repo Class
RFQ Anonymous Not Allowed
RFQ Anonymous Member Not Allowed
RFQ destination member is Unwelcome
Invalid Spot Price
OtcFill Expired
Invalid Quantity
Invalid Rate
Rfq Reply Too Late
Qty Not Compliant With Member Min Qty
Application not available at this time
Invalid Operator Profile
Invalid Settlement System
Invalid Collaterals Settlement Mode
Not Yet Settled
Not Allowed to Edit Order
Invalid Duration
Invalid ClientID
Invalid Settlement Date

Cannot update, too early
Invalid RFQ action
Invalid RFQ Quote Image
RFQ Invalid Yield
Attention: Accept Not Allowed
There is already an exception. Please restore first
Planning section time earlier than security planning time
Invalid OTC Status
Not Enabled Member
Not Allowed AON
Qty Higher Than Issue Qty
Too many transactions per time unit

www.list-group.com