QuantReplay FIX Rules of Engagement

QuantReplay FIX Rules of Engagement

Table of Contents

Introduction

Purpose

This document provides the interface and general functions of the QuantReplay Market Simulator as used through the Financial Information eXchange (FIX) protocol.

General Format

FIX messages are constructed by stringing together field identifiers (called tags) and values. Each pair is separated by a special character (ASCII code 0x01). The messages are communicated in a string format.

The FIX protocol is a series of message standards or definitions. Each message within the FIX protocol is comprised of required, optional and conditionally required fields. Conditionally required fields are those fields that are required if another field in the messages needs further information. For example, options orders require fields to hold put/call codes and limit orders require fields to hold prices.

Each field is composed of 4 parts:

1. Tag Number
2. “=”
3. Field Value
4. Field Delimiter

Each field uses a “<tag>=<value><delimiter>” format. The <tag> attribute represents the field identifier, which is the tag number. The <value> attribute represents the field content or data. The <delimiter> attribute is used to signal the end of the field. For example, the first field in every FIX message is the BeginString field which would be constructed as “8=FIX4.2” where “8” is the tag number representing the BeginString field and “FIX.4.2” is the BeginString field value.

Every FIX message is composed of the message header, message body and message trailer. All messages must have a header and a trailer. Most messages have a body.

A FIX session is identified by a unique FIX Parties pair in the FIX Message Header:

SenderCompID (Tag 49) + TargerCompID (Tag 56)

Time Priority

New resting orders will be entered into the order queue by time priority (first in, first to match) within the range of orders at the same price as each other. Amended orders will lose time priority if the price is amended to any new value, moving to the end of the queue for the range of orders at the new price. Amended orders will also lose time priority if the quantity is amended to a larger value, moving to the end of the queue for the range of orders at the same existing price. Orders that are not amended for price, but amended to a smaller quantity or for any other value of the order will not lose time priority, maintaining its current position in the queue.

Auto Cancel On Disconnect

On termination of a session, either through connection disconnect or session log out:

- All active market data subscriptions are automatically terminated, and require new subscriptions on subsequent successful login.
- If CancelOrdersOnDisconnect is True on the venue, all active orders from a given session are automatically canceled, and on subsequent successful login the new session will receive execution report messages for cancelled

orders. Otherwise all resting orders will remain active even on session disconnect.

Instrument Resolution

The simulator allows clients to specify the listing in their requests using a security identifier or a symbol and a set of additional identification fields. The system uses several algorithms to find the correct listing for a client request. The simulator offers several algorithms for a listing resolution. The concrete algorithm is chosen based on a specified set of instrument identification attributes. The system rejects a request when an algorithm for listing resolution cannot be determined based on a given instrument attribute.

All algorithms search for a listing with attributes equal to those specified in the instrument. Each algorithm also checks for a proper listing party by matching the listing's PartyID and PartyRole values with all groups specified in the request in Parties list. If any listing attribute differs from the one specified in a request, the listing no longer participates in the resolution process. The system rejects requests with an appropriate reject message when resolving a single unique listing is impossible. The reject message type depends on the request message type.

The list of algorithms for the listing resolution supported by the Market Simulator:

Search Algorithm	Priority	Condition	Required Instrument Attributes	Additional Instrument Attributes
Lookup by CUSIP	1	SecurityIDSource (22) = CUSIP	• SecurityID (48)	• Symbol (55) • SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group
Lookup by SEDOL	1	SecurityIDSource (22) = SEDOL	• SecurityID (48)	• Symbol (55) • SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group
Lookup by ISIN	1	SecurityIDSource (22) = ISIN	• SecurityID (48) • SecurityType (167) • SecurityExchange (207) • Currency (15)	• Symbol (55) • Party Group
Lookup by RIC	1	SecurityIDSource (22) = RIC	• SecurityID (48)	• Symbol (55) • SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group

Search Algorithm	Priority	Condition	Required Instrument Attributes	Additional Instrument Attributes
Lookup by Exchange Symbol	1	SecurityIDSource (22) = ExchangeSymbol	• SecurityID (48)	• Symbol (55) • SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group
Lookup by Bloomberg Symbol	1	SecurityIDSource (22) = BloombergSymbol	• SecurityID (48)	• Symbol (55) • SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group
Lookup by Symbol	2	SecurityIDSource (22) is missing in a message AND Symbol (55) is present in a message	• Symbol (55)	• SecurityType (167) • SecurityExchange (207) • Currency (15) • Party Group

Common Components

Standard Header

Tag	Field Name	Req'd	Comments
8	BeginString	Y	Identifies beginning of new message and protocol version. ALWAYS FIRST FIELD IN MESSAGE. (Always sent unencrypted.) Valid values: FIXT.1.1
9	BodyLength	Y	Message length, in bytes, forward to the CheckSum field. ALWAYS SECOND FIELD IN MESSAGE. (Always sent unencrypted.)
35	MsgType	Y	Message length, in bytes, forward to the CheckSum field. ALWAYS THIRD FIELD IN MESSAGE. (Always sent unencrypted.)
1128	ApplVerID	Y	Indicates application version using a service pack identifier. The ApplVerID applies to a specific message occurrence. Supported values: 9 – FIX50SP2
49	SenderCompID	Y	Assigned value used to identify firm sending message.
56	TargetCompID	Y	Assigned value used to identify receiving firm.
34	MsgSeqNum	Y	Integer message sequence number.
50	SenderSubID	N	Assigned value used to identify specific message originator (desk, trader, etc.)

Tag	Field Name	Req'd	Comments
57	TargetSubID	N	Assigned value used to identify specific individual or unit intended to receive message.
43	PossDupFlag	C	Always required for retransmitted messages, whether prompted by the sending system or as the result of a resend request.
97	PossResend	C	Required when message may be duplicate of another message sent under a different sequence number.
52	SendingTime	Y	Time of message transmission.
122	OrigSendingTime	C	Required for message resent as a result of a ResendRequest. If data is not available set to same value as SendingTime.

Standard Trailer

Tag	Field Name	Req'd	Comments
10	CheckSum	Y	Message length. ALWAYS LAST FIELD IN MESSAGE. (Always sent unencrypted.)

Parties

Tag	Field Name	Req'd	Comments
453	NoPartyIDs	Y	Number of party entries.
⇒ 448	PartyID	Y	Party identifier/code.

Tag	Field Name	Req’d	Comments
⇒ 447	PartyIDSource	Y	Identifies class or source of the PartyID (448) value. • D - Proprietary
⇒ 452	PartyRole	Y	Identifies the type or role of the PartyID (448) specified. • 1 - Executing Firm • 17 - Contra Firm

Instrument

Tag	Field Name	Req’d	Comments
55	Symbol	C	Configured data source symbol - Ticker symbol. Required when SecurityIDSource is not set.
48	SecurityID	C	Security identifier value of SecurityIDSource (22) type. Requires SecurityIDSource.

Tag	Field Name	Req'd	Comments
22	SecurityIDSource	C	Identifies the class or source of the SecurityID(48) value. Required if SecurityID (48) is specified. • 1 - CUSIP • 2 - SEDOL • 4 - ISIN • 5 - RIC • 8 - Exchange Symbol • A - Bloomberg Symbol

<table>
<tr><th>Tag</th><th>Field Name</th><th>Req'd</th><th>Comments</th></tr>
<tr><td>167</td><td>SecurityType</td><td>C</td><td>Identifies the class of an asset:
<table>
<tr><th colspan="2">Asset Class</th></tr>
<tr><td>CS - Common Stock (Equity)
FUT - Future
OPT - Option
MLEG - Multi-Leg Instrument
SML - Synthetic Multi-Leg Instrument
WAR - Warrant
MF - Mutual Fund
CORP - Corporate Bond
CB - Convertible Bond
REPO - Repurchase Agreement
INDEX - Index</td><td>CFD - Contract For Difference
CD - Certificate
FXSPOT - Forex Spot
FORWARD - Forward
FXFWD - Forex Forward
FXNDF - Forex Non-Deliverable Forward
FXSWAP - Forex Swap
FXNDS - Forex Non-Deliverable Swap
Required when SecurityIDSource = ISIN</td></tr>
</table></td></tr>
<tr><td>207</td><td>SecurityExchange</td><td>C</td><td>A code of a market where security is listed.

Used to identify the security. Required when SecurityIDSource = ISIN</td></tr>
</table>

Administrative Messages

Heartbeat (35=0)

During periods of inactivity, each party should send heartbeat messages to monitor the status of the communication.

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = 0
112	TestReqID	C	Conditionally required when the heartbeat is the result of a TestRequest message, identifier included in TestRequest message to be returned in resulting Heartbeat. Most FIX clients will utilize a timestamp as this identifier.
Component	Standard Trailer	Y	

Logon

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = 0

Tag	Field Name	Req’d	Comments
98	EncryptMethod	Y	(Always sent unencrypted) 0 - None / Other
108	HeartBtInt	Y	Heartbeat interval (seconds). Note same value used by both sides.
1137	DefaultApplVerID	Y	The default version of FIX being carried over this FIXT session. 9 - FIX50SP2
Component	Standard Trailer	Y	

Logout

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 5
Component	Standard Trailer	Y	

TestRequest

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 1

Tag	Field Name	Req’d	Comments
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat.
Component	Standard Trailer	Y	

ResendRequest

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 2
7	BeginSeqNo	Y	Message sequence number of the first message in range to be resent
16	EndSeqNo	Y	Message sequence number of the last message in range to be resent.
Component	Standard Trailer	Y	

Reject

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 3
45	RefSeqNum	Y	MsgSeqNum - sequence number of rejected message.

Tag	Field Name	Req'd	Comments
58	Text	N	Free text message to explain the reason of the rejection.
371	RefTagID	N	A tag number of the FIX field being referenced (optional). It is included only if a specific tag is the reason for the rejection.
372	RefMsgType	N	The MsgType of the FIX message being referenced.
373	SessionRejectReason	N	Code to identify reason for a session-level Reject message. • 11 - Invalid MsgType
Component	Standard Trailer	Y	

SequenceReset

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = 4
36	NewSeqNo	Y	New sequence number.

Tag	Field Name	Req’d	Comments
Component	Standard Trailer	Y	

Pre-Trade Messages

MarketDataRequest (35=V)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = V
262	MDReqID	Y	Unique identifier for Market Data Request, or the ID of previous Market Data Request to disable if SubscriptionRequestType (263) = 2 (Disable previous Snapshot + Updates Request).
263	SubscriptionRequestType	Y	Subscription Request Type. • 0 - Snapshot • 1 - Snapshot + Updates (Subscribe) • 2 - Disable previous Snapshot + Update Request (Unsubscribe)
264	MarketDepth	Y	Depth of market for Book Snapshot / Incremental updates. • 0 - full book depth • 1 - top of book
265	MDUpdateType	C	Required if SubscriptionRequestType (263) = 1 (Snapshot + Updates) • 0 - Full Refresh • 1 - Incremental Refresh

Tag	Field Name	Req'd	Comments
266	AggregatedBook	N	Specifies whether or not book entries should be aggregated. • Y - book entries to be aggregated (default, and only supported value)
146	NoRelatedSym	Y	Number of symbols (instruments) requested.
⇒ Component	Instrument	Y	
⇒ 15	Currency	C	For non-FX instruments, identifies the currency used for price. Absence of this field is interpreted as the default for the security. For FX instruments, identifies the currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
267	NoMDEntryTypes	Y	Number of MDEntryType fields requested.

Tag	Field Name	Req'd	Comments
⇒ 269	MDEntryType	Y	Must be the first field in this repeating group. This is a list of all the types of Market Data Entries that the firm requesting the Market Data is interested in receiving. • 0 - Bid • 1 - Offer • 2 - Trade • 7 - Trading Session High Price • 8 - Trading Session Low Price • H - Mid Price • b - Market bid • c - MarketOffer • Q - Auction Clearing Price • P - Early Prices
⇒ Component	Parties	N	
Component	Standard Trailer	Y	

MarketDataRequestReject (35=Y)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = Y

Tag	Field Name	Req'd	Comments
262	MDReqID	Y	request ID from rejected request (from rejected request's tag 262).
281	MDReqRejReason	N	codified reason for the rejection of a Market Data request, chosen from one of the values listed: • 0 - Unknown symbol • 1 - Duplicate MDReqID
58	Text	N	free text message describing the reason for the rejection
10	Standard Trailer	Y	

MarketDataSnapshotFullRefresh (35=W)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = W
262	MDReqID	C	Unique identifier for Market Data Request. Conditionally required if this message is in response to a MarketDataRequest(35=V).
Component	Instrument	Y	
779	LastUpdateTime	Y	Timestamp of last update to data item (or creation if no updates made since creation).
268	NoMDEntries	N	Number of entries in Market Data message.

Tag	Field Name	Req'd	Comments
⇒ 269	MDEntryType	Y	Type of market data entry. • 0 - Bid • 1 - Offer • 2 - Trade • 7 - Trading Session High Price • 8 - Trading Session Low Price • H - Mid Price • b - Market Bid • c - Market Offer • Q - Auction Clearing Price • P - Early Price
⇒ 270	MDEntryPx	C	Price of the Market Data Entry.
⇒ 15	Currency	N	Can be used to specify the currency of the quoted price.
⇒ 271	MDEntrySize	C	Quantity or volume represented by the Market Data Entry. MDEntrySize=Indicative Volume.
⇒ 272	MDEntryDate	C	The date the trade happened. Required when MDEntryType=Trade.
⇒ 273	MDEntryTime	C	The time the trade happened. Required when MDEntryType=Trade.
⇒ 288	MDEntryBuyer	C	To report the Buy/Bid order party ID. Required when MDEntryType=Trade.
⇒ 289	MDEntrySeller	C	To report the Sell/Offer order party ID. Required when MDEntryType=Trade.

Tag	Field Name	Req'd	Comments
⇒ 2446	AggressorSide	C	To report side of trades: • 1 (Buy) when the aggressive order was a Buy against a resting Offer • 2 (Sell) when the aggressive order was a Sell against a resting Bid Required when MDEntryType=Trade.
⇒ 336	TradingSessionID	C	Can be used to represent a specific market trading session (e.g. "PRE-OPEN"). To specify good for session where session spans more than one calendar day, use TimeInForce <59> = 'Day' in conjunction with TradingSessionID <336>. • 1 - Day Required when MDEntryType=Trade.
⇒ 625	TradingSessionSubID	C	Optional market assigned sub identifier for a trading session. Usage is determined by market or counterparties. • 2 - Opening Auction for PreOpen • 3 - Continuous Trading • 4 - Closing Auction for PreClosed • 5 - Post-Trading • 6 - Intraday Auction • 9 - Unscheduled Intraday Auction • 10 - Out Of Main Session Trading Required when MDEntryType=Trade.

Tag	Field Name	Req'd	Comments
⇒ 326	SecurityTradingStatus	C	Identifies the trading status applicable to the transaction. • 2 - Trading Halt (Uncrossing of the Auction) • 3 - Resume (Start of the Auction) Required when MDEntryType=Trade.
⇒ 277	TradeCondition		Side of the imbalance. • P = Imbalance more buyers • Q = Imbalance more sellers
278	MDEntryID	Y	MDEntryID value is unique per Listing. This is an incrementing counter of the format timestamp:counter (e.g. 12346523:7) The timestamp component is a Unix Timestamp (Sec) that is generated when the simulator instance starts. The counter component starts from 1 and is incremented separately for each new bid and each new offer.
Component	Standard Trailer	Y	

MarketDataIncrementalRefresh (35=X)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = X

Tag	Field Name	Req'd	Comments
262	MDReqID	C	Unique identifier for Market Data Request. Conditionally required if this message is in response to a MarketDataRequest(35=V).
268	NoMDEntries	N	Number of entries in Market Data message.
⇒ 279	MDUpdateAction	Y	Type of Market Data update action. • 0 - New • 1 - Change • 2 - Delete
⇒ 269	MDEntryType	Y	Type of market data entry. • 0 - Bid • 1 - Offer • 2 - Trade • 7 - Trading Session High Price • 8 - Trading Session Low Price • H - Mid Price • b - Market Bid • c - Market Offer • Q - Auction Clearing Price • P - Early Price
⇒ 270	MDEntryPx	C	Price of the Market Data Entry. MDEntryPx=Indicative Price.
⇒ 15	Currency	C	Can be used to specify the currency of the quoted price. Required when MDEntryType=Trade.

Tag	Field Name	Req’d	Comments
⇒ 271	MDEntrySize	C	Quantity or volume represented by the Market Data Entry. MDEntrySize=Indicative Volume
⇒ 272	MDEntryDate	C	The date the trade happened. Required when MDEntryType=Trade.
⇒ 273	MDEntryTime	C	The time the trade happened. Required when MDEntryType=Trade.
⇒ 277	TradeCondition		Side of the imbalance. • P = Imbalance more buyers • Q = Imbalance more sellers
⇒ 288	MDEntryBuyer	C	To report the Buy/Bid order party ID. Required when MDEntryType=Trade.
⇒ 289	MDEntrySeller	C	To report the Sell/Offer order party ID. Required when MDEntryType=Trade.
⇒ 336	TradingSessionID	C	Can be used to represent a specific market trading session (e.g. "PRE-OPEN"). To specify good for session where session spans more than one calendar day, use TimeInForce <59> = 'Day' in conjunction with TradingSessionID <336>. • 1 - Day Required when MDEntryType=Trade

Tag	Field Name	Req'd	Comments
⇒ 625	TradingSessionSubID	C	Optional market assigned sub identifier for a trading session. Usage is determined by market or counterparties. • 2 - Opening Auction for PreOpen • 3 - Continuous Trading • 4 - Closing Auction for PreClosed • 5 - Post-Trading • 6 - Intraday Auction • 9 - Unscheduled Intraday Auction • 10 - Out Of Main Session Trading Required when MDEntryType=Trade.
⇒ 326	SecurityTradingStatus	C	Identifies the trading status applicable to the transaction. • 2 - Trading Halt (Uncrossing of the Auction) • 3 - Resume (Start of the Auction) Required when MDEntryType=Trade.
⇒ 2446	AggressorSide	C	To report side of trades: • 1 (Buy) when the aggressive order was a Buy against a resting Offer • 2 (Sell) when the aggressive order was a Sell against a resting Bid Required when MDEntryType=Trade.
Component	Standard Trailer	Y	

Trading Messages

NewOrderSingle (35=D)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = D
11	ClOrdID	Y	Unique identifier of the order as assigned by institution.
21	HandlInst	N	Instructions for order handling. • 1 - Automated execution order, private, no Broker intervention • 2 - Automated execution order, public, Broker intervention OK • 3 - Manual order, best execution
38	OrderQty	Y	Quantity ordered.
40	OrdType	Y	Order type. • 1 - Market • 2 - Limit
44	Price	C	Required for OrdType Limit.
54	Side	Y	Side of order. • 1 - Buy • 2 - Sell • 5 - SellShort • 6 - SellShortExempt

Tag	Field Name	Req'd	Comments
⇒ Component	Instrument	Y	
15	Currency	C	For non-FX instruments, identifies currency used for price. Absence of this field is interpreted as the default for the security. For FX instruments, identifies currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
60	TransactTime	Y	Time this order request was initiated/released.
59	TimeInForce	N	Specifies how long the order remains in effect. Absence of this field is interpreted as Day. • 0 - Day • 1 - Good Till Cancel • 3 - Immediate Or Cancel • 4 - Fill Or Kill • 6 - Good Till Date
432	ExpireDate	C	Date of order expiration (last day the order can trade), always expressed in terms of the local market date. Conditionally required if TimeInForce (59) = GTD and ExpireTime (126) is not specified.

Tag	Field Name	Req’d	Comments
126	ExpireTime	C	Time/Date of order expiration (always expressed in UTC). Conditionally required if TimeInForce (59) = GTD and ExpireDate (432) is not specified.
Component	Parties	N	
1688	ShortSellExemption Reason	N	Indicates the reason a short sale order is exempted from applicable regulation. Any integer value is accepted only when Side (54) = SellShortExempt
Component	Standard Trailer	Y	

ExecutionReport (35=8)

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 8
37	OrderID	Y	OrderID is required to be unique for each chain of orders.
11	ClOrdID	N	Required when referring to orders electronically submitted over FIX or otherwise assigned a ClOrdID(11).
41	OrigClOrdID	N	Conditionally required for response to a Cancel or Cancel/Replace request.
17	ExecID	Y	Unique identifier of execution message. ExecutionReport ExecID values are unique per listing per order. (Format:OrderID-Counter)

Tag	Field Name	Req'd	Comments
150	ExecType	Y	Describes the purpose of the execution report. • 0 - New • 4 - Canceled • 5 - Replaced • 8 - Rejected • F - Trade
39	OrdStatus	Y	Describes the current state of the order. • 0 - New • 1 - Partially filled • 2 - Filled • 4 - Canceled • 5 - Replaced • 8 - Rejected
31	LastPx	C	Price of this (last) fill. Required if ExecType (150) = Trade
32	LastQty	C	Quantity bought/sold on this (last) fill. Required if ExecType (150) = Trade
54	Side	Y	Side of order. • 1 - Buy • 2 - Sell • 5 - SellShort • 6 - SellShortExempt

Tag	Field Name	Req'd	Comments
⇒ Component	Instrument	Y	
15	Currency	C	For non-FX instruments, identifies currency used for price. Absence of this field is interpreted as the default for the security. For FX instruments, identifies currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
58	Text	N	Free format text string.
151	LeavesQty	Y	Quantity open for further execution.
14	CumQty	Y	Currently executed quantity for chain of orders.
Component	Parties	N	
1688	ShortSellExemptionReason	N	Indicates the reason a short sale order is exempted from applicable regulation. The exact value is reported as specified in NewOrderSingle/OrderCancelReplaceRequest in ShortSellExemptionReason (1688) for orders with Side (54) = SellShortExempt.
Component	Standard Trailer	Y	

OrderCancelReplaceRequest (35=G)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = G
11	ClOrdID	Y	Unique identifier of replacement request.
37	OrderID	N	Unique identifier of most recent order as assigned.
41	OrigClOrdID	N	ClOrdID(11) of the previous non rejected order (NOT the initial order of the day) when canceling or replacing an order.
21	HandlInst	N	Instructions for order handling. • 1 - Automated execution order, private, no Broker intervention • 2 - Automated execution order, public, Broker intervention OK • 3 - Manual order, best execution
38	OrderQty	Y	Quantity ordered.
40	OrdType	Y	Order type. • 1 - Market • 2 - Limit
44	Price	C	Required for OrdType Limit.
54	Side	Y	Side of order. • 1 - Buy • 2 - Sell • 5 - SellShort • 6 - SellShortExempt

Tag	Field Name	Req’d	Comments
⇒ Component	Instrument	Y	
15	Currency	C	For non-FX instruments, identifies currency used for price. Absence of this field is interpreted as the default for the security. For FX instruments, identifies currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
60	TransactTime	Y	Time this order request was initiated/released.
Component	Parties	N	
1688	ShortSellExemptionReason	N	Indicates the reason a short sale order is exempted from applicable regulation. Any integer value is accepted only when Side (54) = SellShortExempt.
Component	Standard Trailer	Y	

OrderCancelRequest (35=F)

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = F
11	ClOrdID	Y	Unique ID of cancel request.

Tag	Field Name	Req’d	Comments
37	OrderID	N	Unique identifier of most recent order as assigned.
41	OrigClOrdID	N	ClOrdID(11) of the previous non rejected order (NOT the initial order of the day) when canceling or replacing an order.
54	Side	Y	Side of order. • 1 - Buy • 2 - Sell • 5 - SellShort • 6 - SellShortExempt
⇒ Component	Instrument	Y	
15	Currency	C	For non-FX instruments, identifies currency used for price. Absence of this field is interpreted as the default for the security. For FX instruments, identifies currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
60	TransactTime	Y	Time this order request was initiated/released.
Component	Parties	N	

Tag	Field Name	Req’d	Comments
Component	Standard Trailer	Y	

OrderCancelReject (35=9)

Tag	Field Name	Req’d	Comments
Component	Standard Header	Y	MsgType = 9
37	OrderID	Y	OrderID is required to be unique for each chain of orders.
11	ClOrdID	Y	Required when referring to orders electronically submitted over FIX or otherwise assigned a ClOrdID(11).
41	OrigClOrdID	N	ClOrdID(11) which could not be canceled/replaced.
39	OrdStatus	Y	Describes the current state of the order. • 0 - New • 1 - Partially filled • 2 - Filled • 4 - Canceled • 5 - Replaced • 8 - Rejected
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to. • 1 - Order cancel request • 2 - Order cancel/replace request
58	Text	N	Free format text string.

Tag	Field Name	Req'd	Comments
Component	Standard Trailer	Y	

Phase Reporting Messages

Support subscriptions to phase changes reported per security using SecurityStatus workflows when the phase change happens. The SecurityStatusRequest and SecurityStatus messages are both expected to be sent in the context of a specific indicated listing. Changes to phase occur across the entire venue. In any case, the phase will always be reported by listing. This means if a phase change happens across the venue, a separate SecurityStatus message is sent for every listing on that venue (but, of course, only for the client-subscribed listings). If there are any application-layer problems handling the SecurityStatusRequest, the simulator responds with a BusinessMessageReject (35=j) message, indicating the rejection details in Text and BusinessRejectReason tags, and considers the request invalid and terminated.

SecurityStatusRequest request (35=e)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = e
324	SecurityStatusReqID	Y	Unique request ID used to request a snapshot, subscribe to updates, or unsubscribe from a previous subscription that used the same ID
⇒ Component	Instrument	Y	

Tag	Field Name	Req'd	Comments
15	Currency	C	For non-FX instruments, identifies the currency used for price. The absence of this field is interpreted as the default for the security. For FX instruments, identifies the currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
263	SubscriptionRequestType	Y	Subscription Request Type. • 0 - Snapshot • 1 - Snapshot + Updates (Subscribe) • 2 - Disable previous Snapshot + Update Request (Unsubscribe)
Component	Standard Trailer	Y	

SecurityStatus response (35=f)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = e
324	SecurityStatusReqID	N	ID of request that triggered this response
⇒ Component	Instrument	Y	

Tag	Field Name	Req'd	Comments
15	Currency	C	For non-FX instruments, identifies the currency used for price. The absence of this field is interpreted as the default for the security. For FX instruments, identifies the currency used for quantity. Required when SecurityIDSource = ISIN in the Instrument component.
336	TradingSessionID	N	Trading session identifier
625	TradingSessionSubID	N	Sub-identifier for a trading phase within a trading session. • 3 - Open • 10 - Closed
326	SecurityTradingStatus	N	Trading status. • 2 - Trading Halt • 3 - Resume
Component	Standard Trailer	Y	

BusinessMessageReject (35=j)

Tag	Field Name	Req'd	Comments
Component	Standard Header	Y	MsgType = j
45	RefSeqNum	N	The sequence number of the rejected message

Tag	Field Name	Req'd	Comments
372	RefMsgType	Y	The rejected message type
380	BusinessRejectReason	Y	Codified reason for the rejection. • 0 - Other • 1 - Unknown ID • 2 - Uknown Security
58	Text	N	Free text message describing the reason for the rejection
Component	Standard Trailer	Y	

Message Flows

Market Data Streaming

To subscribe to a market data stream, unsubscribe from a previously-subscribed market data stream or retrieve the actual market data state (snapshot) a client shall send the **MarketDataRequest** message. Depending on **SubscriptionRequestType** value, the market simulator:

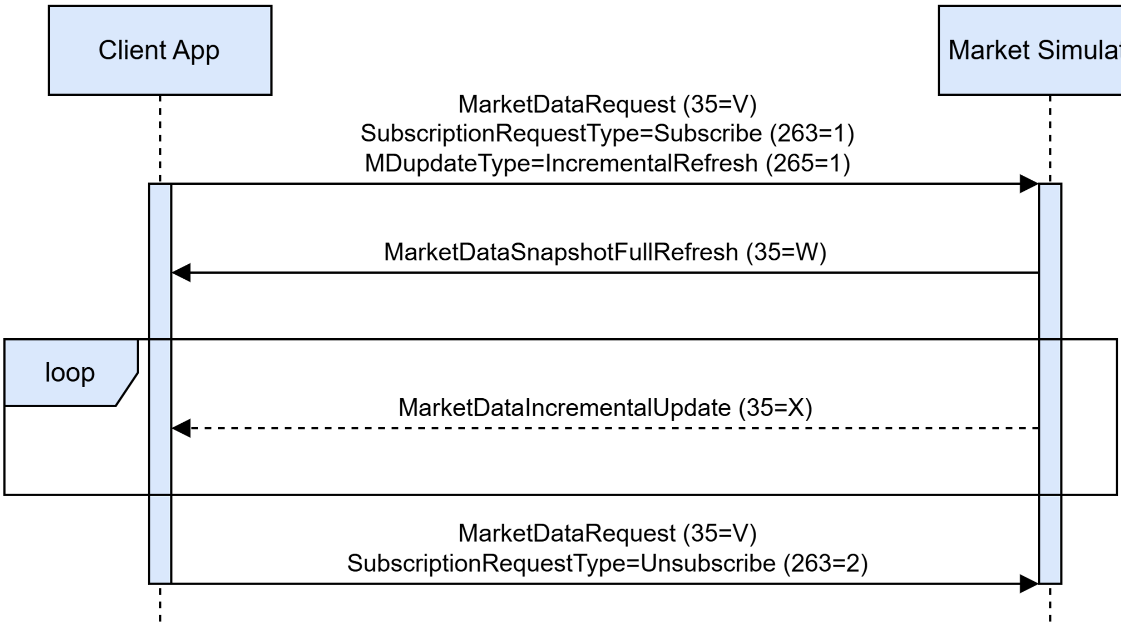
- creates a new subscription and starts requested market data streaming;
- cancels an existing subscription and stops market data streaming for cancelled subscription;
- responds with an actual state of requested market data in the form of a snapshot.

Each market data request can be associated with only one listing. Market data updates are sent for each visible market data change.

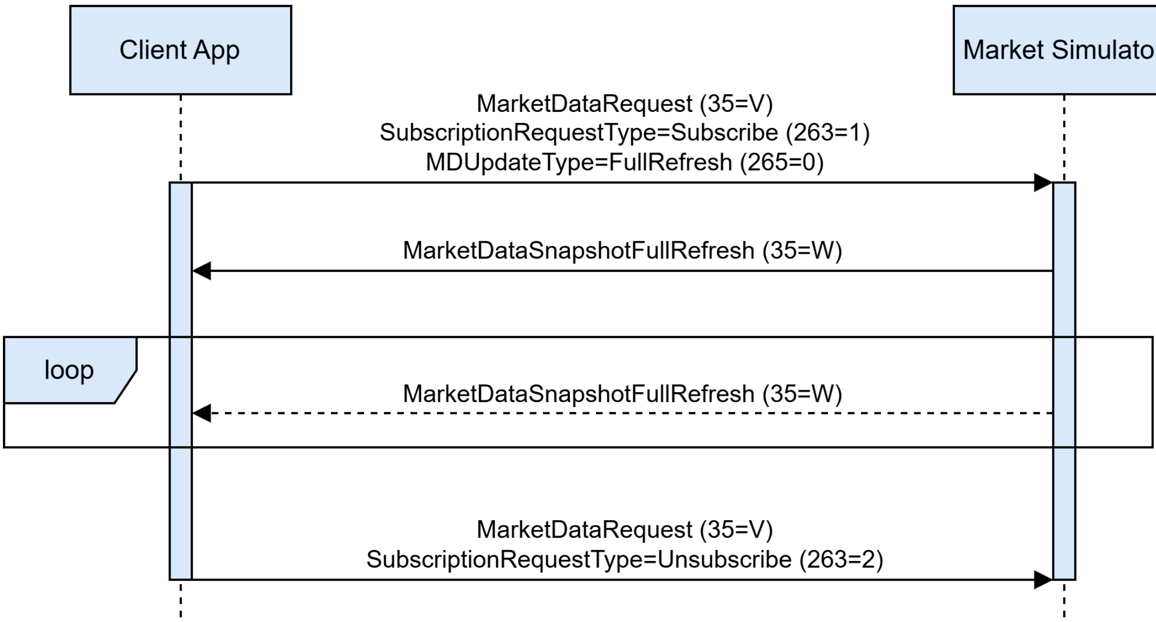
The format of a market data being streamed can be configured by a client in the **MDUpdateType** tag in the **MarketDataRequest** with **SubscriptionRequestType=Subscribe**, which allows to:

- stream the data in the form of incremental updates applied to the initially reported full state;
- stream the data in the form of full market data snapshots.

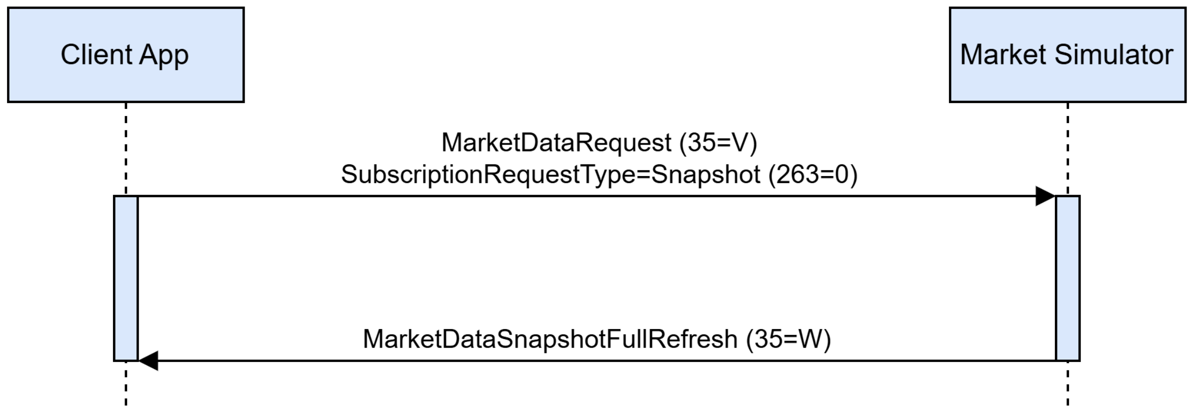
Active market data subscriptions are **automatically cancelled** by the simulator on client disconnection. The market data incremental streaming messages flow between the client application and the simulator:



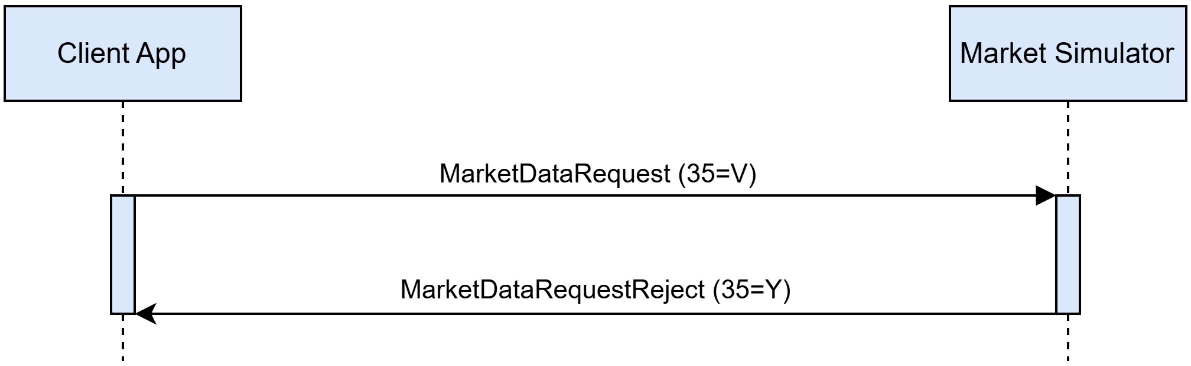
The market data full snapshot streaming messages flow between the client application and the simulator:



The market data snapshot request-response messages flow between the client application and the simulator:



The simulator can reject the Market Data Request message issued by a client due to different circumstances. In case session-level rules violation occurs, the Market Data Request message is rejected by the **Reject** message with the reject reason indicated in the **Text** tag. For application-level rules violation, the **MarketDataRequestReject** message is used to report the rejection to a client, with an optional **MDReqRejReason** and **Text** specifying the details:



The application-level reasons for a MarketDataReject message are listed in the table below with details:

Reason	Condition	MDReqRejReason (tag 281) value	Text (tag 58) value
MarketDataRequest contains no securities inside NoRelatedSym (146) group	–	–	

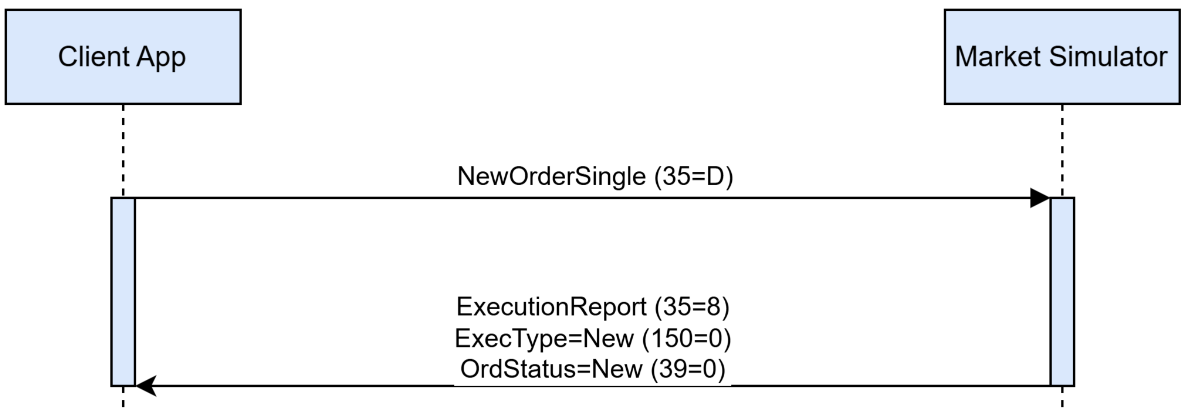
Reason	Condition	MDReqRejReason (tag 281) value	Text (tag 58) value
MarketDataRequest contains more than one security inside NoRelatedSym (146) group	–	–	‘market data requests on multiple securities are now allowed'
No security matches listing identification attributes specified in NoRelatedSym (146) group	–	Unknown Symbol (0)	‘listing not found'
Security identification attributes cannot in NoRelatedSym (146) cannot be used to identify the listing (see Instrument Resolution section)	–	Unknown Symbol (0)	‘listing identification attributes set is malformed’
Subscription MarketDataRequest has MDReqID (262) value is used by another active subscription	MarketDataRequest has SubscriptionRequestType=Subscribe specified	Duplicate MDReqID (1)	'request identifier is used already'

Reason	Condition	MDReqRejReason (tag 281) value	Text (tag 58) value
No active subscription associated with a given MDReqID (262) exists	MarketDataRequest has SubscriptionRequestType=Unsubscribe specified	–	‘no subscription found for the request id’

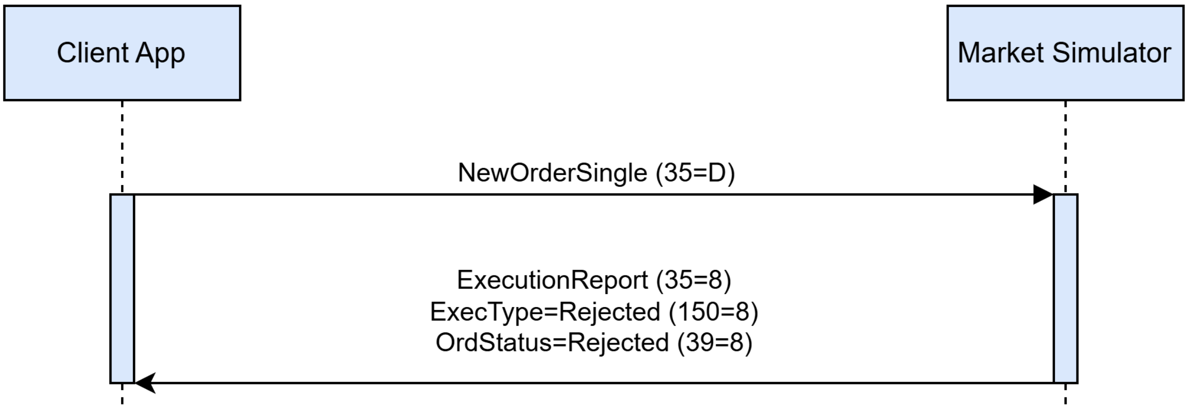
Trading Messages

New Order Single

To create a new order a client shall issue the **NewOrderSingle** message, which should contain new order properties. The simulator responds with an **ExecutionReport** message indicating that an order for a requested listing is created and is ready to be executed. Each successfully created new order is acknowledged with an ExecutionReport containing **ExecType**=**New** and **OrdStatus**=**New**:



A NewOrderSingle message can be rejected by the simulator. An application-level reject is always issued in the form of an **ExecutionReport** message with **ExecType**=**Rejected**, **OrdStatus**=**Rejected**, and a **Text** field specifying the reason for a new order rejection:



NewOrderSingle message rejection reasons are listed in the table below:

Reason	**Condition**	**Text (tag 58) value**
No listing matches listing identification attributes specified in NewOrderSingle	–	'listing not found'
Listing identification attributes cannot be used to identify the listing (see Instrument Resolution section)	–	'listing identification attributes set is malformed'
Price (44) is missing in NewOrderSingle request	Reported when an order with OrdType=Limit is specified without a price	'order price not found'
Price (44) is specified in NewOrderSingle request	Reported when an order with OrdType=Market is specified with a price	'order price is not allowed'
OrderQty (38) is missing in NewOrderSingle request	–	'order quantity missing'

Reason	Condition	Text (tag 58) value
Value in Price (44) is not a multiple of the requested listing's price tick	–	'order price tick constraint is violated'
Value in OrderQty (38) is not a multiple of the requested listing's quantity multiple	–	'order quantity multiple constraint is violated'
Value in OrderQty (38) is less than the requested listing's quantity minimum	–	'minimal order quantity constraint is violated'
Value in OrderQty (38) is greater than the requested listing's quantity maximum	–	'maximal order quantity constraint is violated'
An opposite side has no orders matching the new order to be traded	Reported only for orders with OrdType=Market or TimeInForce=ImmediateOrCancel or TimeInForce=FillOrKill	'no facing orders found'
All matching opposite orders cannot fill a new FillOrKill order	Reported only for orders with TimeInForce=FillOrKill	'not enough liquidity to fill FoK order'
Both ExpireDate (432) and ExpireTime (126) are missing for a GoodTillCancel order	Reported only for orders with TimeInForce=GoodTillCancel	'neither expire date nor expire time specified'

<table>
<tr><th>Reason</th><th>Condition</th><th>Text (tag 58) value</th></tr>
<tr><td>Both ExpireDate (432) and ExpireTime (126) are specified for a GoodTillCancel order</td><td>Reported only for orders with TimeInForce=GoodTillCancel</td><td>‘both expire date and expire time specified’</td></tr>
<tr><td>A new GoodTillCancel order is already expired as per the specified ExpireDate (432) or ExpireTime (126)</td><td>Reported only for orders with TimeInForce=GoodTillCancel</td><td>‘order already expired’</td></tr>
<tr><td>The venue is in the Open halted phase.

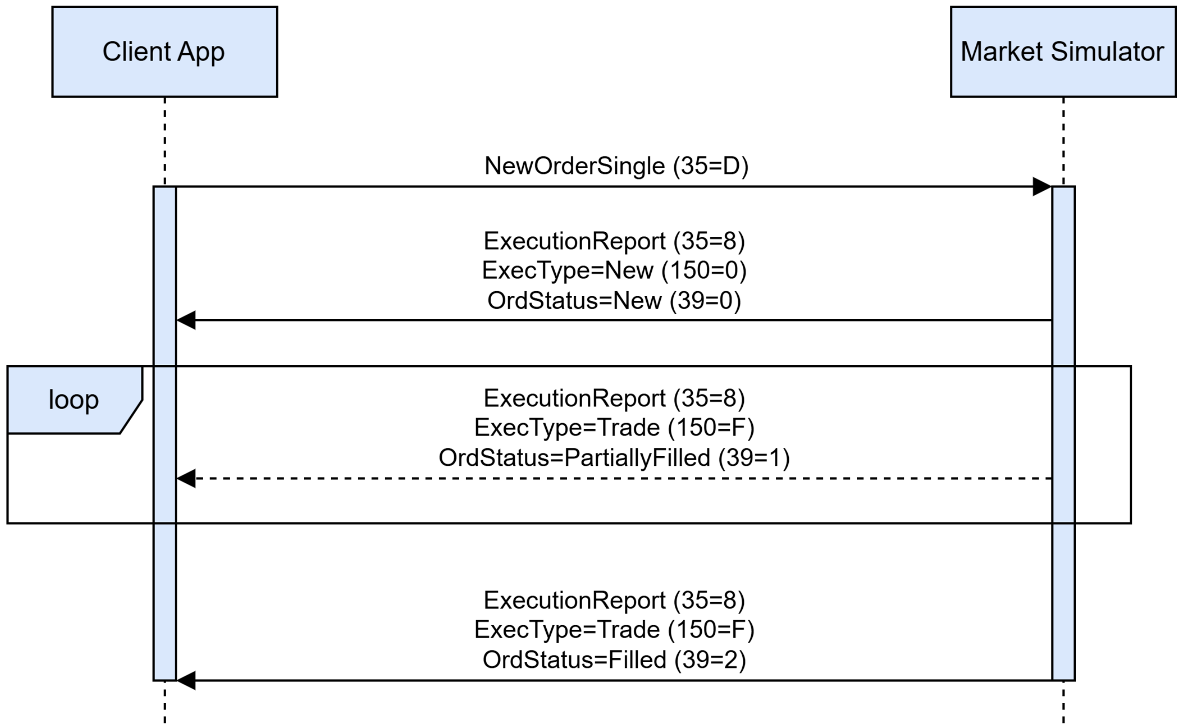
It means that in the message SecurityStatus (35=f), the tags are:

• TradingSessionSubID is “(Continuous) Trading” - 625=3

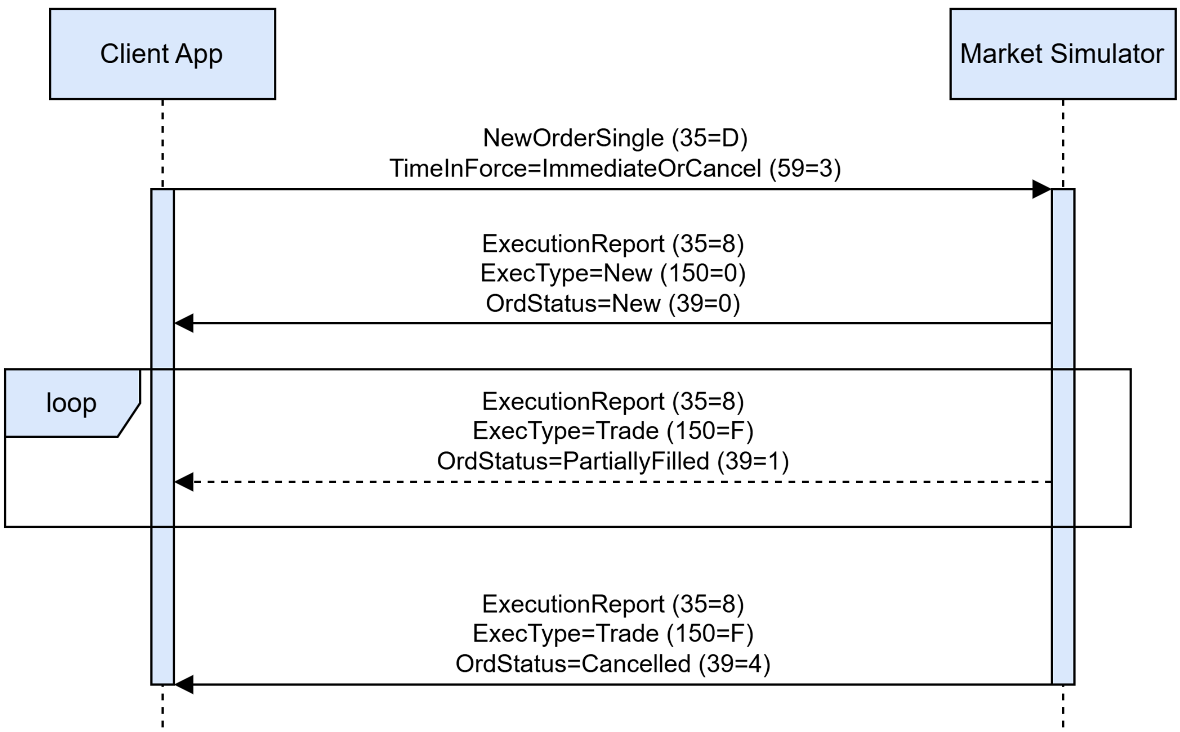
• SecurityTradingStatus is “Trading halt” - 326=2</td><td>–</td><td>‘request cannot be processed during halted trading status’</td></tr>
</table>

Execution Report

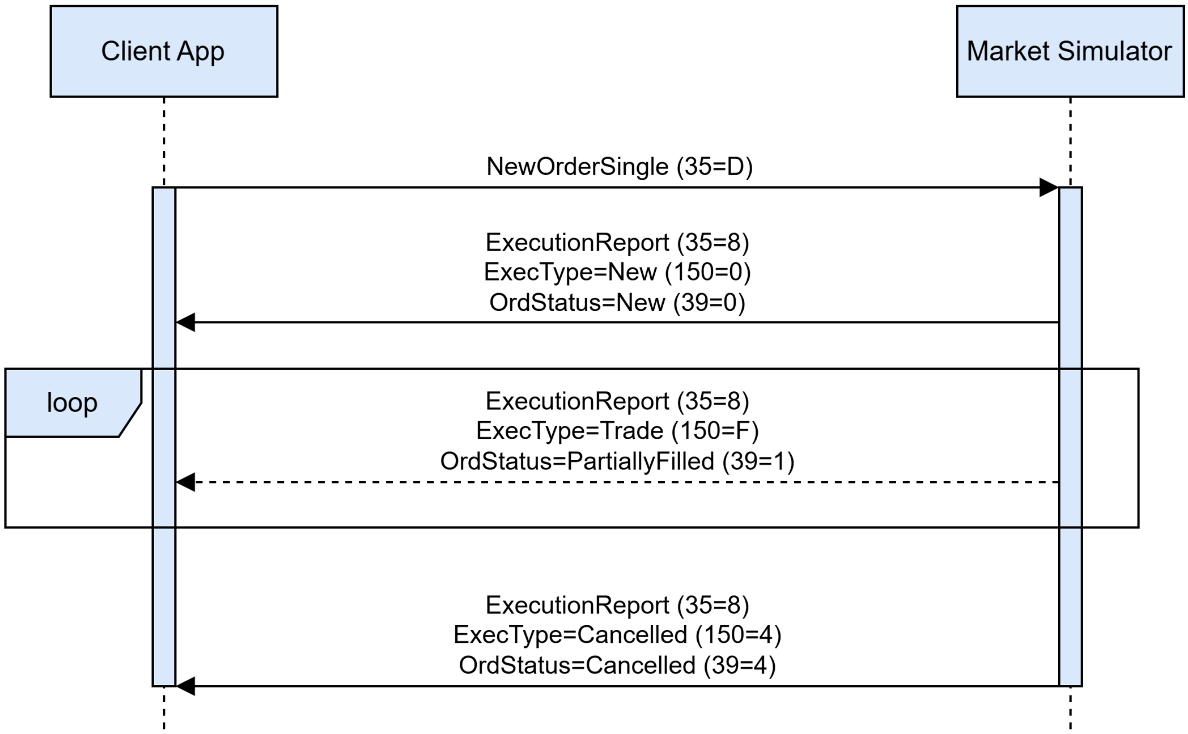
Once a NewOrderSingle message is accepted and acknowledged by the simulator, an order can be traded. Each trade event is reported by the **ExecutionReport** message with **ExecType**=**Trade**. The order status is changed to **OrdStatus**=**PartiallyFilled** if there is volume left for being executed, otherwise order status is changed to **OrdStatus**=**Filled** and the order is terminated. The diagram below illustrates the basic message flow for the order lifetime between a client and the simulator:



The simulator cancels orders with **TimeInForce**=**ImmediateOrCancel** automatically if there is a non-executed order part left after the order matching and trading process. Such cancellation would be reported by **OrdStatus**=**Cancelled** in the last execution report for an order:

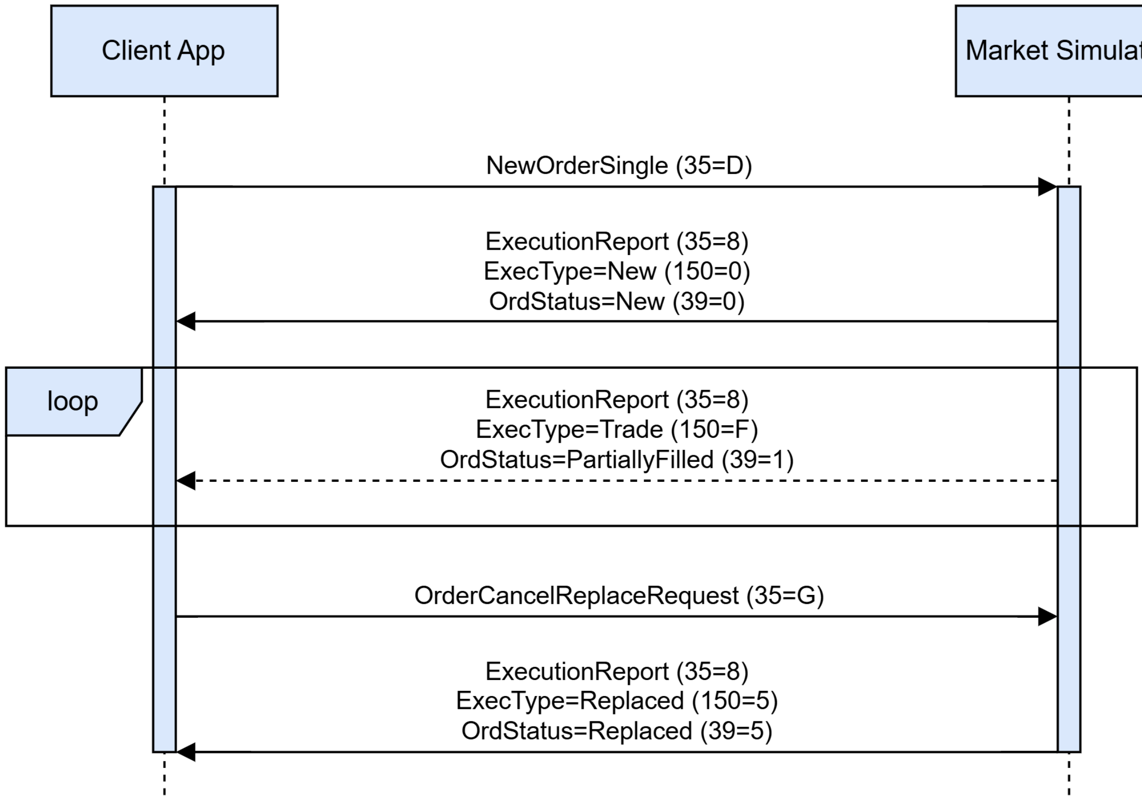


The simulator automatically cancels expired orders with **TimeInForce**=**Day** and **TimeInForce**=**GoodTillDate**. The cancellation is reported to a client in the form of an ExecutionReport message with **ExecType**=**Cancelled** and **OrdStatus**=**Cancelled**:

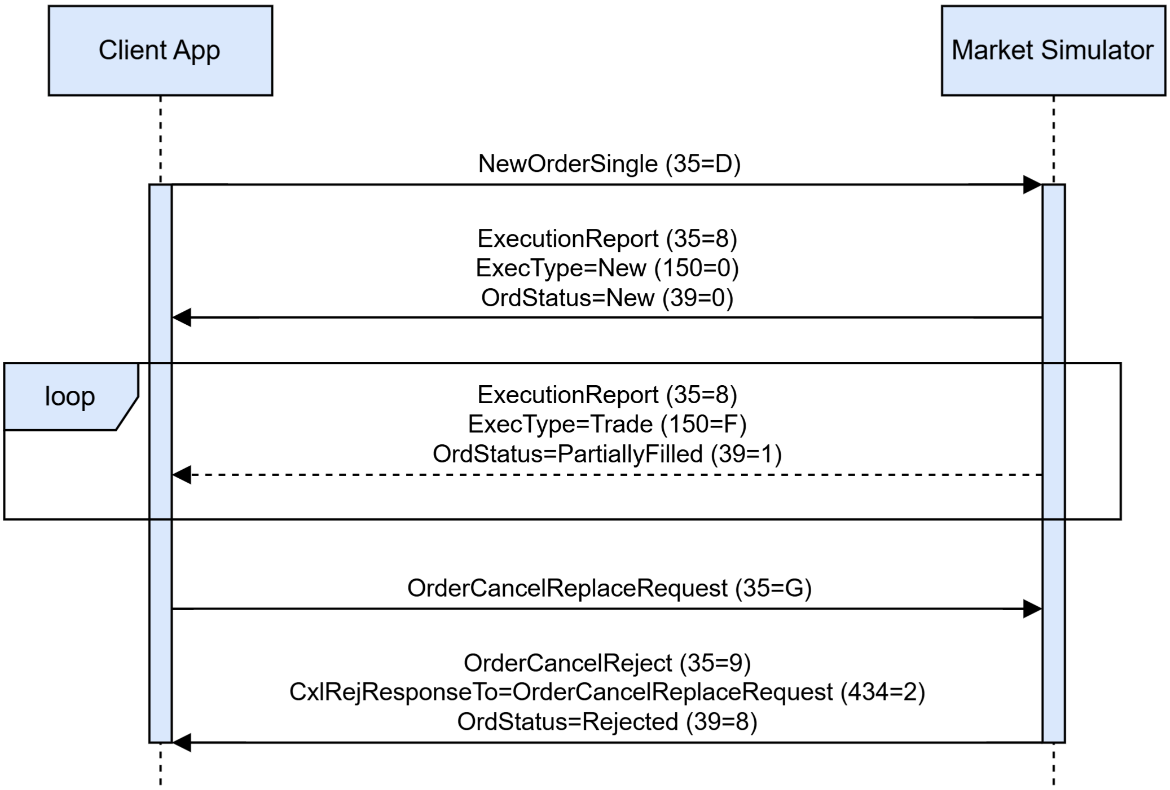


Order Cancel Replace Request

A client may issue an **OrderCancelReplaceRequest** to modify his active order. A successful modification is reported by an ExecutionReport message with **ExecType**=**Replaced** and **OrdStatus**=**Replaced**, as shown in the diagram:



An OrderCancelReplaceRequest message can be rejected by the simulator. An application-level reject is always issued in the form of an **OrderCancelReject** message with **CxlRejResponseTo=OrderCancelReplaceRequest**, **OrdStatus=Rejected**, and a Text field specifying the reason for a modification rejection. An order is not modified if OrderCancelReplaceRequest is rejected.



OrderCancelReplaceRequest message rejection reasons are listed in the table below:

Reason	**Condition**	**Text (tag 58) value**
No listing matches listing identification attributes specified in OrderCancelReplaceRequest	–	‘listing not found'
Listing identification attributes cannot be used to identify the listing (see Instrument Resolution section)	–	‘listing identification attributes set is malformed’

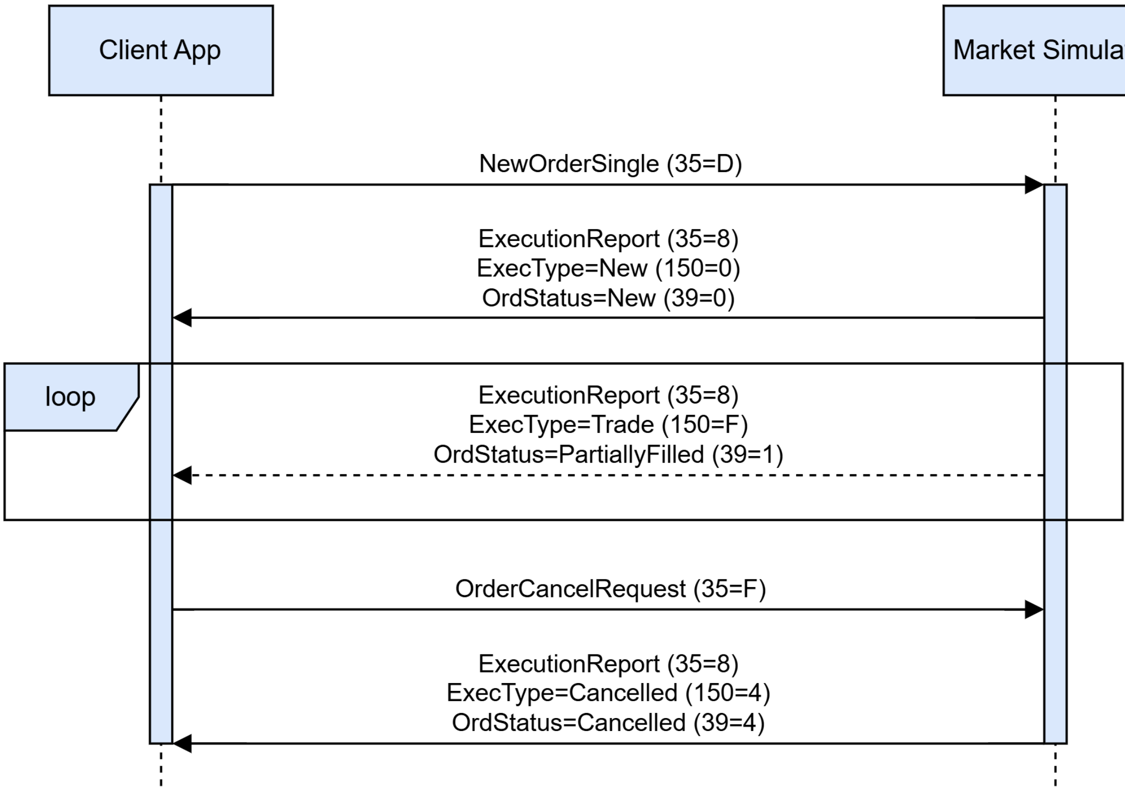
Reason	Condition	Text (tag 58) value
Given OrigClOrdID (41), ClOrdID (11), or OrderID (37) do not match any client order for a requested listing. Side (54) does not match the original order side value. Order is terminated (e.g. filled or cancelled)	–	‘order not found’
New OrderQty (38) value is less than or equal to the already executed CumQty (14) for an order	–	‘invalid quantity’
New TimeInForce (59) value does not match the original value specified for an order	–	‘time in force value can not be changed’
Price (44) is missing in NewOrderSingle request	Reported when an order with OrdType=Limit is specified without a price	‘order price not found’
OrderQty (38) is missing in NewOrderSingle request	–	‘order quantity missing’
Value in Price (44) is not a multiple of the requested listing’s price tick	–	‘order price tick constraint is violated’
Value in OrderQty (38) is not a multiple of the requested listing’s quantity multiple	–	‘order quantity multiple constraint is violated’

Reason	Condition	Text (tag 58) value
Value in OrderQty (38) is less than the requested listing's quantity minimum	–	'minimal order quantity constraint is violated'
Value in OrderQty (38) is greater than the requested listing's quantity maximum	–	'maximal order quantity constraint is violated'
Both ExpireDate (432) and ExpireTime (126) are missing for a GoodTillCancel order	Reported only for orders with TimeInForce=GoodTillCancel	'neither expire date nor expire time specified'
Both ExpireDate (432) and ExpireTime (126) are specified for a GoodTillCancel order	Reported only for orders with TimeInForce=GoodTillCancel	'both expire date and expire time specified'
A new GoodTillCancel order is already expired as per specified ExpireDate (432) or ExpireTime (126)	Reported only for orders with TimeInForce=GoodTillCancel	'order already expired'

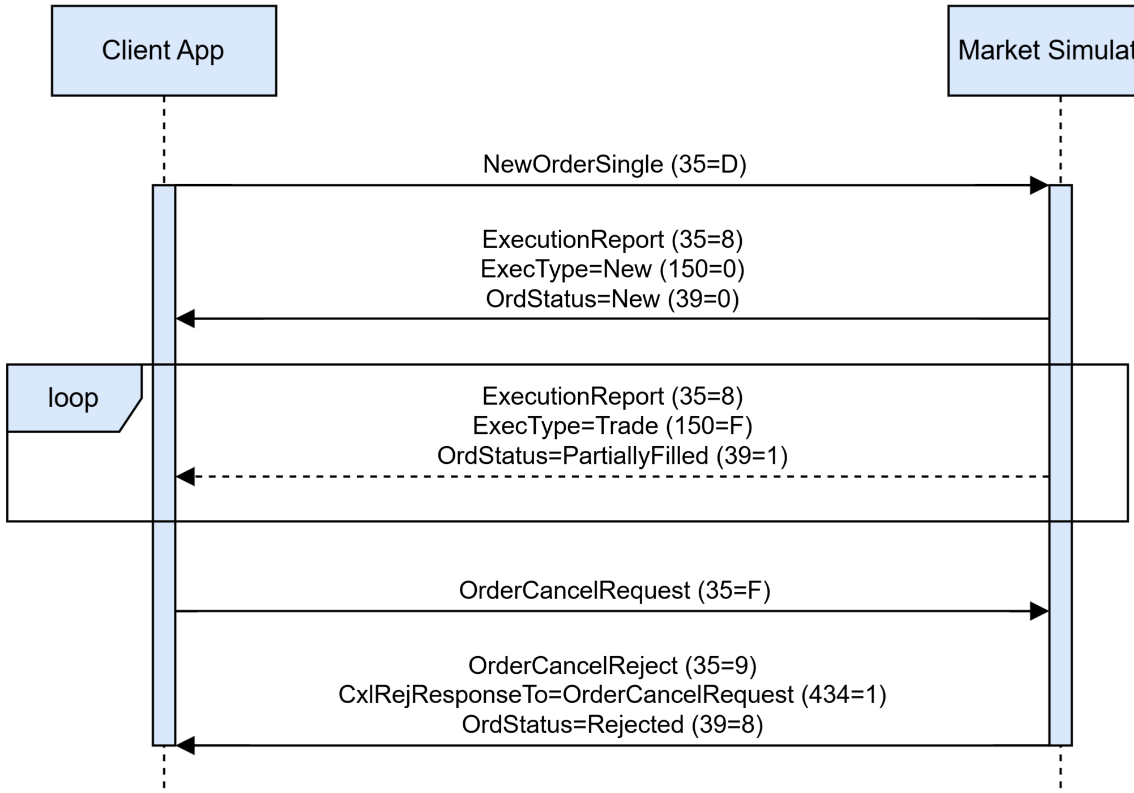
Reason	Condition	Text (tag 58) value
The venue is in the Open halted phase. It means that in the message SecurityStatus (35=f),the tags are: • TradingSessionSubID is “(Continuous) Trading” - 625=3 • SecurityTradingStatus is “Trading halt” - 326=2	–	‘request cannot be processed during halted trading status'

Order Cancel Request

A client may issue an **OrderCancelRequest** (35=F) to cancel his active order. A successful cancellation is reported by an ExecutionReport message with **ExecType**=**Cancelled** and **OrdStatus**=**Cancelled**, as shown in the diagram:



An OrderCancelReplaceRequest message can be rejected by the simulator. An application-level reject is always issued in the form of an **OrderCancelReject** message with **CxlRejResponseTo**=**OrderCancelRequest**, **OrdStatus**=**Rejected**, and a **Text** field specifying the reason for a cancellation rejection. An order remains active if OrderCancelRequest is rejected.



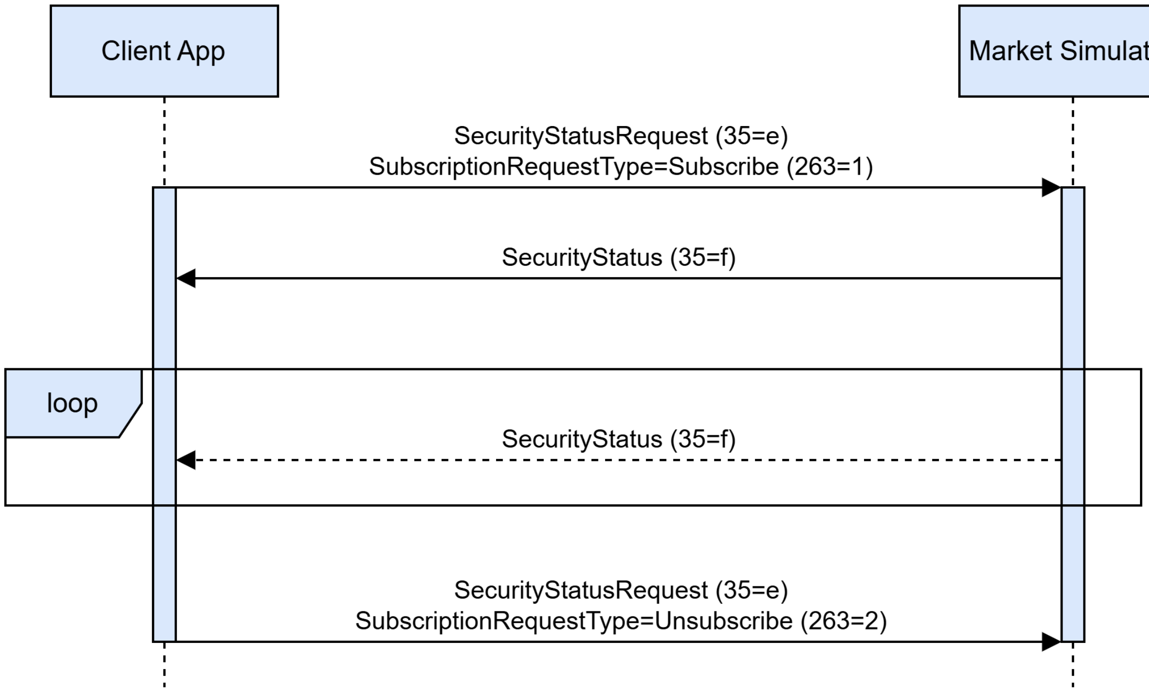
OrderCancelRequest message rejection reasons are listed in the table below:

Reason	Condition	Text (tag 58) value
No listing matches listing identification attributes specified in OrderCancelRequest	–	‘listing not found'
Listing identification attributes cannot be used to identify the listing (see Instrument Resolution section)	–	‘listing identification attributes set is malformed’

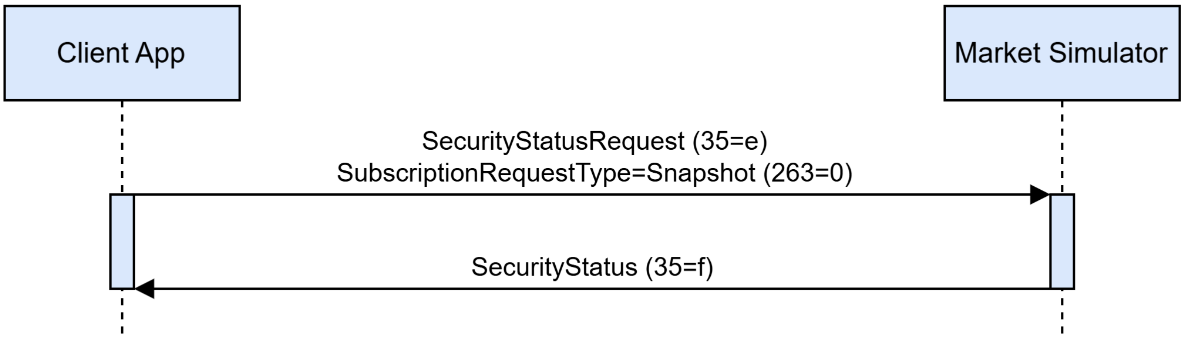
Reason	Condition	Text (tag 58) value
Given OrigClOrdID (41), ClOrdID (11), or OrderID (37) do not match any client order for a requested listing. Side (54) does not match the original order side value. Order is terminated (e.g. filled or cancelled)	–	‘order not found’
The venue is in the Open halted phase and OrderCancelRequest is not allowed on this halted phase. “Open halted phase” means that in the message SecurityStatus (35=f), the tags are: • TradingSessionSubID is “(Continuous) Trading” - 625=3 • SecurityTradingStatus is “Trading halt” - 326=2	–	‘request cannot be processed during halted trading status'

Trading Phase and Trading Status Reporting

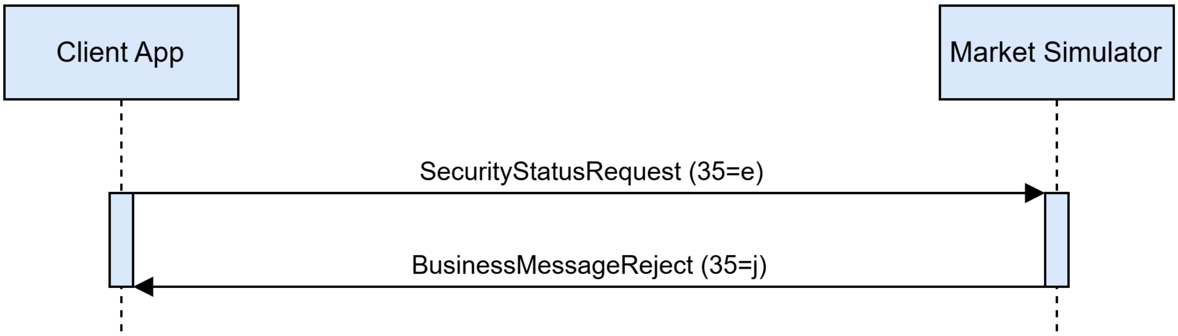
The Market Simulator allows clients to retrieve trading phase and status updates for a specific security by issuing a **SecurityStatusRequest** message with the **SubscriptionRequestType=Subscribe**. The simulator creates a subscription for a client and starts updates streaming in the form of **SecurityStatus messages**, each defining a currently active trading phase and its status. The subscription is considered to be active until a client sends an unsubscribe request in the form of **SecurityStatusRequest** message with **SubscriptionRequestType=Unsubscribe**. A subscription is automatically cancelled upon client disconnect. The general trading phase/status reporting message flow is described in a diagram below:



A client may also issue a current trading phase/status information only, without the creation of a subscription and further updates streaming. In this case, a client shall send a **SecurityStatusRequest** message with **SubscriptionRequestType=Snapshot**. The simulator replies with the **SecurityStatus** message only, as shown in the diagram below:



A SecurityStatusRequest message can be rejected by the simulator. An application-level reject is always issued in the form of a **BusinessMessageReject** message with **BusinessRejectReason** and **Text** specifying reject details.



SecurityStatusRequest message application level reject reasons listed in a table below.

Reason	**Condition**	**BusinessRejectReason (tag 380) value**	**Text (tag 58) value**
No listing matches listing identification attributes specified in NewOrderSingle	–	Unknown Security (2)	‘listing not found'
Listing identification attributes cannot be used to identify the listing (see Instrument Resolution section)	–	Unknown Security (2)	‘listing identification attributes set is malformed’

Reason	Condition	BusinessRejectReason (tag 380) value	Text (tag 58) value
SecurityStatusRequest has SecurityStatusReqID (324) value used by another active subscription	SecurityStatusRequest is sent with SubscriptionRequestType=Subscribe	Other (0)	'security status subscription request id is already in use'
No active subscription associated with a given SecurityStatusReqID (324) exists	SecurityStatusRequest is sent with SubscriptionRequestType=Unsubscribe	Unknown ID (1)	‘no subscription found for the security status request id’