NSE ORDER ROUTING, QUOTE MANAGEMENT & MARKET DATA FIX SPECIFICATION

THE Nigerian STOCK EXCHANGE

Version	1.21
Date	18 January 2018
File	NSE Order Routing, Quote Management and Market Data FIX Specification

Document History

VERSION	DATE	SUMMARY OF CHANGES
1.0	2010-06-16	First release of Genium FIX standard requirement specification.
1.1	2010-10-19	Removed fields NextExpectedMsgSeqNum(789), SessionStatus(1409) and TestMessageIndicator(464). Added enumeration(5) for OrdRejReason(103) Added table showing messages supported by the Order Management Gateway and the Market Data Gateway.
1.2	2010-10-20	Further amendments to make clearer that the implementation involves two Gateways. Removed unreferenced Field Enumerations.
1.3	2010-10-22	Additional changes from review of 1.2
1.4	2010-11-22	Client OrderID changes. Change to 6.10 Security Definition Request (c).
1.5	2011-11-23	Added description for mixed FIX and Native protocols. New diagrams for OBO handling in Genium FIX for X-stream.
1.6	2012-02-09	Updated field NetChgPrevDay (451) in MarketDataSnapshotFullRefresh (W).
1.7	2012-05-24	Table 19 - Tag 236 (Yield) updated to 'C'. B.4 TriggeringInstruction Component Block - OrdStatus (39) definition updated to 'X'. MDEntryDate(272) and MDEntryTime(273) added to Table 23 - Order Information Response Tags. 'No Trades Exist' enumeration for MDEntryType(269) added to Appendix C – Field Enumerations. OrdType (40): Invalid ordertype Q= Counter-order selection removed. 'Not supported' removed from tag 925 NewPassword in Table 3 – Logon. Table 45 – Instrument Component Block: REQ'D value and Comments for SecurityIDSource and SecurityID updated. Enumeration '1' added for Tag 102 in Appendix C – Field Enumerations. PositionEffect and QuoteID removed from Table 11 – New Order Single, Table 13 –

VERSION	DATE	SUMMARY OF CHANGES
		Order Cancel/Replace Request, Table 18 – Execution Report. PositionEffect also removed from Appendix C – Field Enumerations. HeartBtInt(108) range note added to section 2.2 Logon (A). Tag 553 updated to mandatory in Table 3 - Logon. Tag 11 comment clarified in Table 15 – Order Status Request. Tag 60 - TransactTime comments updated in Table 17 - Order Mass Action Report. Section 5.2.7 Trade Information Request and Responses - Only last 10 trades reported in MD Snapshot. Section 5.2.8 Index Information Request and Response – Added number of trades, total value and total volume. Section 5.2.9 Security Information Request and Response – Added number of trades and open quantity. B.1 Instrument (symbology) Component Block – Added ProductComplex for Sector Identifier Order Cancel Reject removed from Table 19 - Execution Report Returned Tags Based On Scenario. Added Price Limit in section 6.8 Security List and section 6.9 Security List Update Report. Updated B.2 Parties Component Block. Added MatchType tag in Table 21 - TradeCaptureReport and its enumerations in Appendix C – Field Enumerations. Removed value 6 in tag MassActionScope (1374) in Appendix C – Field Enumerations. Added MarketSegmentID (1300) in Table 21 - TradeCaptureReport (AE) Added SecuritySubType (762) in Table 45 - Instrument Component Block, removed TradingSessionID (336) in Table 11 – New Order Single, Table 18 – Execution Report and Section 5 – Market Data Corrected levels in captions and replaced page breaks with section breaks. Removed Order Manager Gateway and Market Data Gateway sections in Section 1 GENIUM FIX Specification for X-stream Removed Section 3.2.3 Unsolicited Modifications Removed mapping from TradingSessionID (336) to X-stream Board Id in Appendix C. Deleted RawData in Section 2.2 Logon. Removed TradingSessionID (336) from Section 3. Order Management and Section 5. Market Data. Added SecuritySubType (762) in B.1. Instrument Component Block for X-stream Board Id. Corrected MarketSegmentID (1300) comments in Section 5 Market data and Section 6 Reference Data. Added Appendix B.2 Underlying Instrument Component Group. Added AccruedInterestAmt (159) in Table 23 – Trade Capture Report Changed 3.2 Order Modification Updated Section 1.4 FIX Messages Supported Updated Sesion 2.1 FIX Session Establishment to include UserName (553) Updated Table 11 – New Order Single, Price for yield based porducts is dirty price. Removed TradeID (1003) in Section 4.5 Trade Capture Report Reqeust Ack (AQ) Delected Parties component block in Market Data messages.

VERSION	DATE	SUMMARY OF CHANGES
		Added TradingReferencePrice (1150) in SecurityListUpdateReport(BK) message
1.8	2012-06-13	Revert Table 11 - *New Order Single* and Table 13 – *Order Cancel/Replace Request*, Price for yield based products is clean price. Remove OrdStatus (39) – 'W' from B.6 TriggeringInstruction Component Block
1.9	2012-08-29	Removed 'Arabic' replaced with 'non-English', correct FIX code for SecurityDefinitionRequest.
1.10	2013-03-19	Removed Order Replace/Cancel Reject row in Table 18 - Execution Report Returned Tags Based On Scenario. Added "Only snapshot (0) is supported" for tag SubscriptionRequestType (263) in Table 37 – Market Definition Request. Changed tag NoRelatedSym (146) and Instrument block to required in Market Data Request (V). Updated comments accordingly in Table 27 – Market Data Request. Mantis 56685: Updated order/price depth levels in sections 5.2 – 5.5 (Market Data).
1.11	2013-05-21	Added ISIN to Table 45 – Instrument Component Block 58481: PATCH-Updates to FIX Specification - Currency(15) removed from section 5.3 *MarketDataRequest(V)*; Currency(15) for Trade Capture Report removed from Table 27 – *Market Data Request*; MDReportID(963) removed from Table 29 - *Market Data Snapshot/Full Refresh.* 59496: MarketDepth (264) should be MDEntryPositionNo (290) in Table 23 - *Order Information Response Tags* for Market By Price. Figure 9 – *Order Status States* updated.
1.12	2013-06-27	Section 4 *Quote Management* added. Mantis 61571: CR 6 Market Maker and upgrade GPCFIX to BASE.
1.13	2013-12-09	Order handling instructions updated for AON and WON orders in sections 3.7 *New Order Single (D)* and 3.9 *Order Cancel/Replace Request (G).* Tag 18 - ExecInst updated in Appendices C1 and C2.
1.14	2014-10-16	*Table 28 - Security Statistics Returned Tags*, *Table 51 – Field Enumerations Sorted by Tag Value* and *Table 52 – Field Enumerations Sorted By Tag Name* updated. Added possible value for 269-MDEntryType=p for indicative closing price and indicative closing volume. Added 336-TradingSessionID for 269=z and 269=p. *Figure 9 – Order Status States* updated (Reject of acc'd order removed).
1.15	2015-03-26	Table 25 - *Order Information Response Tags*: tag 37-OrderId added. Section 2.2 *FIX Session establishment scenarios* added.
1.16	2016-06-30	Tag 1126 added to Table 23 - *Trade Capture Report*. Document updated from 'NSE OMX' to 'NSE'.
1.17	2016-08-25	Tag 1390 added to Table 23 - *Trade Capture Report*. Added possible value for 59-TimeInForce = 7 for Pre Close in *Table 51 – Field Enumerations Sorted by Tag Value* and *Table 52 – Field Enumerations Sorted By Tag Name*. Description corrected in section 2.2.2 *Reconnect*.

VERSION	DATE	SUMMARY OF CHANGES
1.18	2016-11-02	Added Trade Capture Tags to facilitate Neg Deals Add LotType(1093) for Block Divestment Remove reference to tag 964 from Table 45 – Security List and Table 46 – Security List Update Report
1.19	2017-05-11	*Table 29 - Security Statistics Returned Tags*: Added Official opening and closing flags and imbalance quantity. *Table 32 – Market Data Snapshot/Full Refresh* and *Table 33 – Market Data Incremental Refresh*: Added 286-OpenCloseSettlFlag. *Table 37– Security Status*: Added 1174-SecurityTradingEvent. *Table 56– Field Enumerations Sorted by Tag Value* and *Table 57– Field Enumerations Sorted By Tag Name:* Added A-imbalance value to 269-MDEntryType and 1174-SecurityTradingEvent.
1.20	2017-11-02	Added CR29 Derivatives tags for Instrument block and SecurityDefinition (+ update). Added Tags UnderlyingSecurityID(309), UnderlyingSecurityIDSource(305).
1.21	2018-01-18	Table 37 – Security Status: Added more description for tag 1174. Table 56– Field Enumerations Sorted by Tag Value and Table 57– Field Enumerations Sorted By Tag Name: Added S-Session value to 59-TimeInForce.

Contents

Figures

Tables

1 NSE FIX Specification for X-stream

This document provides the X-stream FIX message specification supporting version 5.0 (SP1) of the FIX protocol specification.

The messages specified in this document support most existing X-stream clients that are using FIX V4.4 engine implementation. There are however areas of exceptions:

- Deprecated FIX V4.4 fields have been replaced with the equivalent (and endorsed) V5.0 fields (e.g. 'MaxFloor' has been replaced with 'DisplayQty')
- User defined V4.4 fields which have standard V5.0 tag equivalents have been replaced with the V5.0 tag values.

1.1 FIX V5.0 (SP1) Supported Messages

The FIX V5.0 (SP1) specific messages are:

- Market Definition Request (BT), Market Definition (BU)
- Trading Session List Update Report (BS)
- Security List Update Report (BK)
- Security Definition Update Report (BP)

It should be noted that some request and/or response messages will contain both FIX V4.4 and V5.0 (SP1) fields. Tag values at 1000 or above are V5.0 (SP1) tags.

FIX V5.0 tag values are all identified within the applicable messages.

1.2 Document Structure

This document is divided into the following chapters:

Chapter 2 – Session and Infrastructure messages.

Chapter 3 – Order Management

Chapter 4 – Trade Capture Reports

Chapter 5 – Market Data

Chapter 6 – Reference Data

Appendix A describes the FIX Standard Header and Standard Trailer.

Appendix B describes the various component blocks used throughout this document.

Appendix C describes the enumerations for fields with multiple valid values.

Appendix D describes the FIX data types referred to in the 'format' column of each message description.

1.3 References

The following references are documents related to this specification.

Table 1 – Reference Documents

DOCUMENT	DESCRIPTION
Volume 1 – Introduction to the FIX Protocol V5.0 (SP1)	Provides information on the FIX protocol including common components, data types and usage.
Volume 2 – Fix Protocol Specification V4.4.	Provides information on session level FIX messages
Volume 3 – FIX Protocol Specification V5.0 (SP1)	Provides information on pre-trade FIX messages
Volume 4 – FIX Protocol Specification V5.0 (SP1)	Provides information on order and execution FIX messages
Volume 5 – FIX Protocol Specification V5.0 (SP1)	Provides information on post-trade FIX messages

1.4 FIX Messages Supported

The following table lists the FIX messages that are supported by the NSE FIX Gateway. NSE FIX Gateway can be configured to support order management (OM) message only, or support both order management (OM) and market data (MD) messages.

Table 2 – FIX Messages Supported

MESSAGE NAME	MESSAGE TYPE	GATEWAY		MESSAGE DIRECTION	MESSAGE FUNCTION
		OM	MD+OM		
Administrative messages					
Logon	A	Y	Y	Inbound Outbound	Identifies and authenticates a user/member establishing a connection to the gateway.
Logout	5	Y	Y	Inbound Outbound	Used to terminate a FIX session.
Reject	3	Y	Y	Inbound Outbound	Response message providing notification regarding messages that cannot be processed by the gateway or FIX Client.
Resend Request	2	Y	Y	Inbound Outbound	Initiates a re-transmission of messages from the gateway.
Sequence Reset (Gap Fill)	4	Y	Y	Inbound Outbound	Message has two modes: Sequence Reset - Gap Fill and Sequence Reset-Reset.
Test Request	1	Y	Y	Inbound Outbound	Verifies sequence numbers or communications line status.
Heartbeat	O	Y	Y	Inbound Outbound	Monitors gateway status during periods of inactivity.
Business Message Reject	j	Y	Y	Outbound	Rejects any application message that cannot be processed by the Gateway and cannot be rejected via another message.

MESSAGE NAME	MESSAGE TYPE	GATEWAY		MESSAGE DIRECTION	MESSAGE FUNCTION
		OM	MD+OM		
Order Management messages					
New Order - Single	D	Y	Y	Inbound	Used by institutions wishing to electronically submit securities orders for execution.
Order Cancel Request	F	Y	Y	Inbound	Request to cancel all of the remaining quantity of an existing order.
Order Cancel / Replace Request	G	Y	Y	Inbound	Request message to change the details of an existing order.
Order Cancel Reject	9	Y	Y	Outbound	Reject message for an Order Cancel / Replace Request or Order Cancel Request that cannot be honoured.
Order Status Request	H	Y	Y	Inbound	Request for querying the details of an order.
Order Mass Action Request	CA	Y	Y	Inbound	Used to request the cancellation or status of a group of orders that match the criteria specified in the request.
Order Mass Action Report	BZ	Y	Y	Outbound	Acknowledement to an Order Mass Action Request.
Execution Report	8	Y	Y	Outbound	Responds with the action X-stream has taken in response to a new or existing order including acknowledges Order Cancel and Cancel / Replace Requests, order history requests and report fills to orders. Fills against orders are reported via the Execution Report message, as are trade cancels.
Trade Capture Reporting messages					
Trade Capture Report Request	AD	Y	Y	Inbound	Request all firm trades and to subscribe or unsubscribe for trade capture reports.
Trade Capture Report	AE	Y	Y	Inbound / Outbound	Responds to a Trade Capture Report Request and used to report matched trades.
Trade Capture Report Request Ack	AQ	Y	Y	Outbound	Used to indicate if no trades matched the selection critiera specified in the Trade Capture Report Request or the Trade Capture Request was invalid.

MESSAGE NAME	MESSAGE TYPE	GATEWAY		MESSAGE DIRECTION	MESSAGE FUNCTION
		OM	MD+OM		
Market Data messages					
Market Data Request	V	N	Y	Inbound	Requests current best market information in a market for a security.
Market Data Request Reject	Y	N	Y	Outbound	Rejects market data request messages that cannot be honoureddue to business or technical reasons.
Market Data Snapshot / Full Refresh	W	N	Y	Outbound	Responds to the Market Data Request message with the current best market information for a security.
Market Data Incremental Refresh	X	N	Y	Outbound	Used for Market Data incremental updates.
Trading Session Status Request	g	Y	Y	Inbound	Request information on the status of a market.
Trading Session Status	h	Y	Y	Outbound	Responds with the current status of a market.
Security Status Request	e	Y	Y	Inbound	Requests the status of a security. One or more Security Status message are returned as a result of a Security Status Request.
Security Status	f	Y	Y	Outbound	Responds with the current state of a security that is currently listed. May be filtered by board.
News	B	Y	Y	Outbound	Contains bulletin messages initiated by the Exchange.
Reference Data messages					
Market Definition Request	BT	Y	Y	Inbound	Request for market structure information from the Exchange.
Market Definition	BU	Y	Y	Outbound	Respond to Market Definition Request.
Trading Session List Request	BI	Y	Y	Inbound	Request a list of trading sessions available in a market placeand the state of those trading sessions.
Trading Session List	BJ	Y	Y	Outbound	Respond to a Trading Session List Request containing the characteristics of the trading session(s).

MESSAGE NAME	MESSAGE TYPE	GATEWAY		MESSAGE DIRECTION	MESSAGE FUNCTION
		OM	MD+OM		
Trading Session List Update Report	BS	Y	Y	Outbound	Provides intra-day updastes of trading sessions when there are changes to one or more trading sessions.
Security List Request	x	Y	Y	Inbound	Requests a list of securities form the Exchange that match criteria provided in the request.
Security List	y	Y	Y	Outbound	Responds wil a list of securites that match the criteria specified in a Security List Request.
Security List Update	BK	Y	Y	Outbound	Responds with updates to the reference database. Updates could be due to Corporate Action or other business events.
Security Definition Request	c	Y	Y	Inbound	Request the definition of a specific security, set of individual securities for a sinlge market segment or all securities.
Security Definition	d	Y	Y	Outbound	Responds to the Security Definition Request.
Security Definition Update	BP	Y	Y	Outbound	Responds with updates to the reference database. Updates could be due to Corporate Action or other business events.

2 Session and Infrastructure Messages

This section defines the FIX Session and Infrastructure messages. This section also describes the FIX Session establishment actions.

The FIX Session Level messages are:

- Logon
- Logout
- Reject
- Resend Request
- Sequence Reset (Gap Fill)
- Test Request
- Heartbeat

The FIX Infrastructure messages are:

- Business Message Reject

2.1 FIX Session Establishment

2.1.1 Logon and Authentication

A FIX session must be established with X-stream before the exchange of business messages is allowed. The session is established using the Logon message and part of session establishment processing includes the authentication of the initiator. This requires that a valid SenderCompID (49) which defines the party initiating the session, and a password, is provided in the Logon message which can be used for security authentication purposes. A FIX session will not be established if authentication processing fails.

2.1.2 Logon Failures and Account Locking

All logon failures return a Logout message with an appropriate reason code and may include additional text which provides additional information regarding the failure. If the session initiator fails to authenticate with the X-stream system within a defined number of attempts, the account will be locked and all subsequent logon attempts will be rejected. If logons are disabled by the marketplace, a failure to logon will not cause the account to be locked but only rejected. Any other causes for authentication failure will cause the account to be locked after a defined number of failed attempts. To unlock the account requires marketplace operations to reset the account and assign a new password.

2.1.3 FIX Session SenderCompID, Username and Passwords

The SenderCompID (49) , username (553) and session password are always required for authentication with X-stream and must be included in the Logon message. Both the SenderCompID, username (553) and session password (when in plain text) can have a maximum length of 32 characters.

2.1.4 Changing FIX Session Passwords

Passwords can be changed using the FIX session Login messages and the following will apply:

- Passwords used for X-stream session authentication must be changed on a periodic basis as they have a limited lifetime. Passwords can be changed programmatically using the Logon message only at session establishment and only while they are valid. If the password has expired or cannot be changed programmatically due to limitations of the Participant's

FIX implementation then they must be changed manually by Business or Technical Operations.

- To change the FIX session password at logon time both the current password and the new password must be included in the FIX Logon message. In addition, the SenderCompID (49), UserName (553) and the current password must be valid otherwise authentication will fail.
- The current password is sent using the Password (554) field in the Logon message. It should be noted that the password will transit external and internal X-stream networks in plain text if encryption is not utilized (refer to Section 2.1.5 regarding encryption).
- Providing the SenderCompID, UserName (553) and current password are valid, the new password is checked against the password policy for compliance. If the new password complies, it is updated in the X-stream database and becomes the password to be used for the next session logon. If the new password does not comply with the password policy then an error status and message is returned in the Logon confirm message. However, regardless of whether the new password complies or not with the password policy the FIX session will be established if the SenderCompID, UserName and existing password are still valid.

2.1.5 Encryption

NSE FIX for X-stream does not currently support either password or message encryption over FIX sessions. If encryption is required then hardware based encryption must be used.

2.1.6 FIX Session Logon Confirmation and Logout

Upon receipt of a Logon message and after successful authentication, a Logon message is returned as an acknowledgement indicating that a session has been established. If a session logon has failed for any reason a FIX Logout message is returned. Both the FIX Logon and Logout messages include fields which are used to return status and text information pertaining to either a successful or failed session logon.

2.1.6.1 Logon Confirmation (Session Authenticated)

Both the Logon and Logout message contain the Text(58) field which provide additional may be returned in the Logon notification message. For example, the text information returned may indicate the application version in use, why the new password did not comply with password policies, the number of days until the password expires or other information deemed relevant by the marketplace.

2.1.6.2 Logout (Authentication Failure)

Failure to establish a session with X-stream for any reason will return a Logout message. The Text(58) field in the Logout message may contain additional useful information regarding the reason for the Logout message being returned.

2.2 FIX Session establishment scenarios

There are four types of FIX session establishment scenarios:

1. At the start of the day, a new FIX session should be used to connect to the FIX server.
2. If for any reason a FIX session is disconnected, the FIX session should try to reconnect to the same FIX server.
3. If the FIX session cannot be reconnected to the same FIX server, it should fail over to the backup FIX server.
4. If the FIX session cannot be established with the backup FIX server, a new FIX session should be used to connect to the disaster recovery FIX server.

2.2.1 Start of the day

A new FIX session is required at the start of the day with MsgSeqNum (34) set to 1 in the Logon message. The FIX session should expect MsgSeqNum (34) in the reply message to start from 1. However, the FIX server might generate execution reports before log on, incrementing the MsgSeqNum (34) to be greater than 1. In this case, the Resend Request message can be used to retrieve messages generated before log on from sequence number 1.

2.2.2 Reconnect

When reconnecting to the same FIX server, the Logon message should set MsgSeqNum (34) to the last MsgSeqNum (34) sent plus 1. A FIX session can only be reconnected in the same day.

The FIX session should also expect the MsgSeqNum (34) from the FIX server to be the last MsgSeqNum (34) received plus 1. Resend Request message can be used to retrieve messages missed during disconnection, if a higher than expected MsgSeqNum (34) is received.

All subscriptions are lost on disconnection and therefore the FIX client must re-subscribe.

2.2.3 Fail over

Connecting to the backup FIX server is similar to reconnect, except that previously submitted FIX message subscriptions will not be kept. The FIX session should subscribe these messages again.

2.2.4 Disaster recovery

Connecting to the disaster recovery server is the same as connecting at the start of the day, except that the FIX server address is different.

2.3 Logon (A)

The logon message authenticates a user establishing a connection to a remote system. The logon message must be the first message sent by the application requesting to initiate a FIX session.

Table 3 – Logon

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = A	
98	EncryptMethod	Y	(Always unencrypted)	Int
108	HeartBtInt	Y	Note same value used by both sides	Int
141	ResetSeqNumFlag	N	Indicates both sides of a FIX session should reset sequence numbers	Boolean
1137	DefaultApplVerID	Y	Specifies the service pack release being applied by default to the message at the session level. The only valid value is '8' = FIX50SP1.	String
553	Username	Y	Specifies a different username or userID to use for authentication	String
554	Password	Y	Note: minimal security exists without transport-level encryption	String
925	NewPassword	N	Specifies a new password when required.	String
58	Text	N	Free format text string	String

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Trailer		Y		

The FIX gateway accepts HeartBtInt(108) range from 10 to 60. If client HeartBtInt is out of this range, the server will reply with the last valid value, or the default value (60) if it is the first logon of the day.

2.4 Logout (5)

The logout message initiates or confirms the termination of a FIX session. Disconnection without the exchange of logout messages should be interpreted as an abnormal condition.

The logout format is as follows.

Table 4 – Logout

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 5	
58	Text	N	Free format text string	String
Standard Trailer		Y		

2.5 Reject (3)

The reject message should be issued when a message is received but cannot be properly processed due to a session-level rule violation. An example of when a reject may be appropriate would be the receipt of a message with invalid basic data (e.g. MsgType=&) which successfully passes de-encryption, CheckSum and BodyLength checks. As a rule, messages should be forwarded to the trading application for business level rejections whenever possible.

Rejected messages should be logged and the incoming sequence number incremented.

The reject format is as follows.

Table 5 – Reject

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 3	
45	RefSeqNum	Y	MsgSeqNum of rejected message	SeqNum
371	RefTagID	N	The tag number of the FIX field being referenced.	Int
372	RefMsgType	N	The MsgType of the FIX message being referenced.	String
373	SessionRejectReason	N	Code to identify reason for a session-level Reject message.	Int
58	Text	N	Free format text string	String
Standard Trailer		Y		

2.6 Resend Request (2)

The resend request is sent by the receiving application to initiate the retransmission of messages. This function is utilized if a sequence number gap is detected, if the receiving application lost a message, or as a function of the initialization process.

The resend request can be used to request a single message, a range of messages or all messages subsequent to a particular message.

The resend request format is as follows.

Table 6 – Resend Request

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 2	
7	BeginSeqNo	Y		SeqNum
16	EndSeqNo	Y		SeqNum
Standard Trailer		Y		

2.7 Sequence Reset (Gap Fill) (4)

The Sequence Reset message has two modes: Gap Fill mode and Reset mode.

Gap Fill mode

Gap Fill mode is used in response to a Resend Request when one or more messages must be skipped over for the following reasons:

During normal resend processing, the sending application may choose not to send a message (e.g. an aged order). During normal resend processing, a number of administrative messages are skipped and not resent (such as Heart Beats, Test Requests). Gap Fill mode is indicated by GapFillFlag (tag 123) field = "Y". If the GapFillFlag field is present (and equal to "Y"), the MsgSeqNum should conform to standard message sequencing rules (i.e. the MsgSeqNum of the Sequence Reset GapFill mode message should represent the beginning MsgSeqNum in the GapFill range because the remote side is expecting that next message sequence number).

Reset mode

Reset mode involves specifying an arbitrarily higher new sequence number to be expected by the receiver of the Sequence Reset-Reset message, and is used to establish a FIX session after an unrecoverable application failure.

Reset mode is indicated by the GapFillFlag (tag 123) field = "N" or if the field is omitted. The Sequence Reset format is as follows.

Table 7 – Sequence Reset

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 4	
123	GapFillFlag	N		Boolean
36	NewSeqNo	Y		SeqNum
Standard Trailer		Y		

2.8 Test Request (1)

The test request message forces a heartbeat from the opposing application. The test request message checks sequence numbers or verifies communication line status. The opposite application responds to the Test Request with a Heartbeat containing the TestReqID.

The TestReqID verifies that the opposite application is generating the heartbeat as the result of Test Request and not a normal timeout. The opposite application includes the TestReqID in

the resulting Heartbeat. Any string can be used as the TestReqID (one suggestion is to use a timestamp string). The test request format is as follows.

Table 8 – Test Request

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 1	
112	TestReqID	Y		String
Standard Trailer		Y		

2.9 Heartbeat (0)

The Heartbeat monitors the status of the communication link and identifies when the last of a string of messages was not received.

When either end of a FIX connection has not sent any data for [HeartBtInt] seconds, it will transmit a Heartbeat message. When either end of the connection has not received any data for (HeartBtInt + "some reasonable transmission time") seconds, it will transmit a Test Request message. If there is still no heartbeat message received after (HeartBtInt + "some reasonable transmission time") seconds then the connection should be considered lost and corrective action be initiated. If HeartBtInt is set to zero then no regular heartbeat messages will be generated. Note that a test request message can still be sent independent of the value of the HeartBtInt, which will force a Heartbeat message.

Heartbeats issued as the result of Test Request must contain the TestReqID transmitted in the Test Request message. This is useful to verify that the Heartbeat is the result of the Test Request and not as the result of a regular timeout.

The heartbeat format is as follows.

Table 9 – Heartbeat

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = 0	
112	TestReqID	N	Required when the heartbeat is the result of a Test Request message.	String
Standard Trailer		Y		

2.10 Business Message Reject (j)

The Business Message Reject message can reject an application-level message which fulfils session-level rules and cannot be rejected via any other means. Note if the message fails a session-level rule (e.g. body length is incorrect), a session-level Reject message should be issued.

Table 10 – Business Message Reject

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = j (lowercase)	
45	RefSeqNum	N	MsgSeqNum of rejected message	SeqNum
372	RefMsgType	Y	The MsgType of the FIX message being referenced.	String
379	BusinessRejectRefID	N	The value of the business-level "ID" field on the	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			message being referenced. Required unless the corresponding ID field (see list above) was not specified.	
380	BusinessRejectReason	Y	Code to identify reason for a Business Message Reject message. Code to identify reason for a Business Message Reject message.	Int
58	Text	N	Free format text string	String
Standard Trailer		Y		

3 Order Management

The order management category consists of the following messages:

- New Order Single
- Order Cancel Request
- Order Cancel Replace Request
- Order Cancel Reject
- Order Status Request
- Order Mass Action Request
- Order Mass Action Report
- Execution Report.

The figures below describe the workflow for new order entry, order cancellation, order modification and order status.

3.1 Unique ClOrderId (11)

X-stream will not check for uniqueness of ClOrdId(11) on New Order Single, Order Cancel/Replace Request and Order Cancel Request messages. Firms submitting order transactions via FIX interface must ensure unique ClOrdId(11) is entered on these transactions.

3.2 Order Modification

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

- Fields not set in the Cancel Replace *will be reset*. To keep the original value, the same field must be set with the same value in the Cancel Replace.
- The required fields must be set regardless if they can be changed or not.

3.2.1 Order Attributes allowed to change

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

- OrderQty (38)
- DisplayQty (111)
- Price (44)
- OrdType (40)
- TimeInForce (59)
- Yield (236)
- ExpireDate (432)
- ExpireTime (126)
- Account (1)

- ExecInst (18)
- TriggerPrice (1102)

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

3.2.2 Order Identification

In an Order Cancel Replace Request, the order can be identified by either its prior ClOrdID using OrigClOrdID (41), or by the OrderID (37). If OrderID is used, OrigClOrdID should be set to "NONE". OrderID (37) is unique for every order. OrderID (37) doesn't change after order amendment.

3.3 Order Cancellation

- If the user wishes to cancel a single previously sent order, the Order Cancel Request message is used.
- Execution Reports are issued relaying the status of every canceled order.
- In some cases orders may be cancelled in the system without prior request by the user. These will be sent as unsolicited Execution Reports to the client.
- The system will generate cancel messages (Execution Report –IOC/Fok Order Cancel) for every IOC and FoK order.

3.3.1 Order Identifcation

In an Order Cancel Request the order can be identified by either it's prior ClOrdID using OrigClOrdID (41), or by the OrderID (37). If OrderID is used, OrigClOrdID should be set to "NONE".

3.4 Cross Protocol Order Management

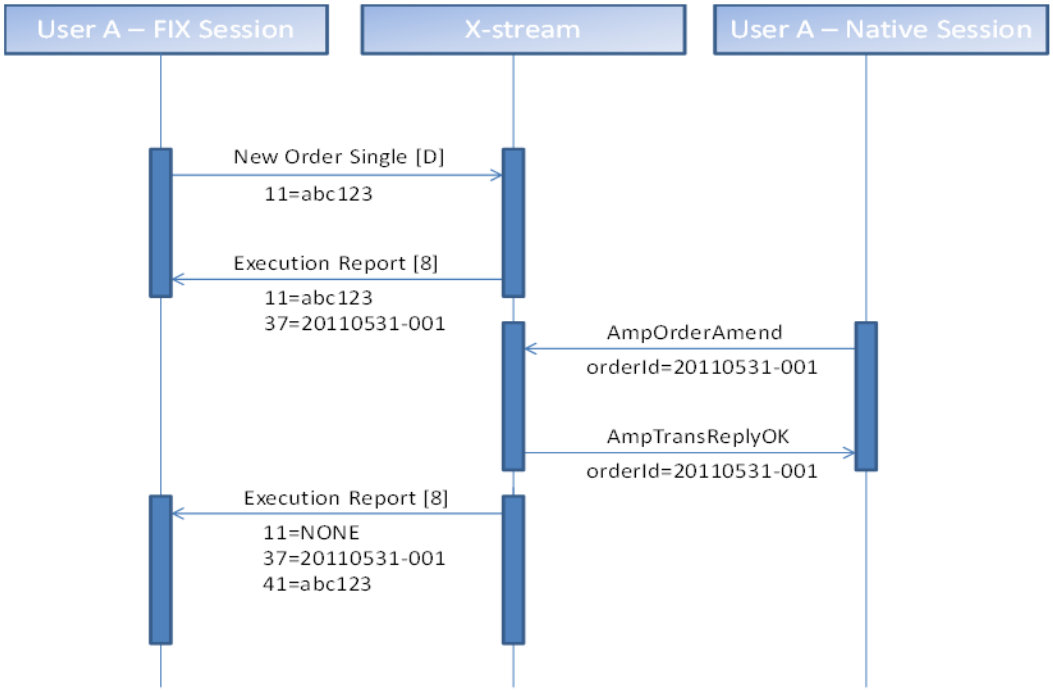
All orders entered via the native protocol will be published as Execution Reports on the FIX session(s) for the same member. Please note that no ClOrdID will be set on those Execution Reports. These orders can be replaced, cancelled or status requested via FIX. System-generated OrderID (37) will be available.

- For a Cancel Request or a Cancel Replace Request, set OrigClOrdID to "NONE", and provide the valid OrderId (37) instead.
- For Order Status Requests, set ClOrdID to "NONE" and supply the OrderID (37).

3.4.1 Orders Entered via FIX

Orders entered via FIX can be altered or cancelled using the native protocol. A FIX client must be prepared to receive unsolicited order updates and cancels.

Figure 1 – FIX Order Amended via Native Protocol

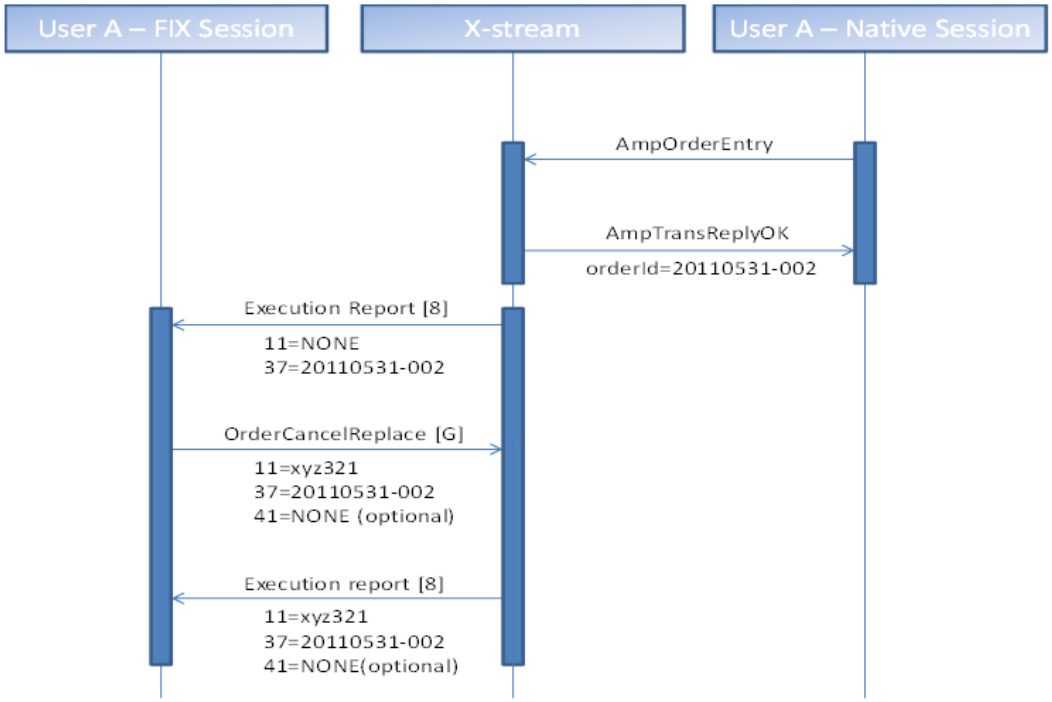


If the order is amended twice by a native client, then 41 will not be present in the ExecutionReport (8)

3.4.2 Orders Entered via Native Protocol

Orders entered via the native protocol may be amended via FIX using tag (37). The user may not be logged on via both protocols at the same time.

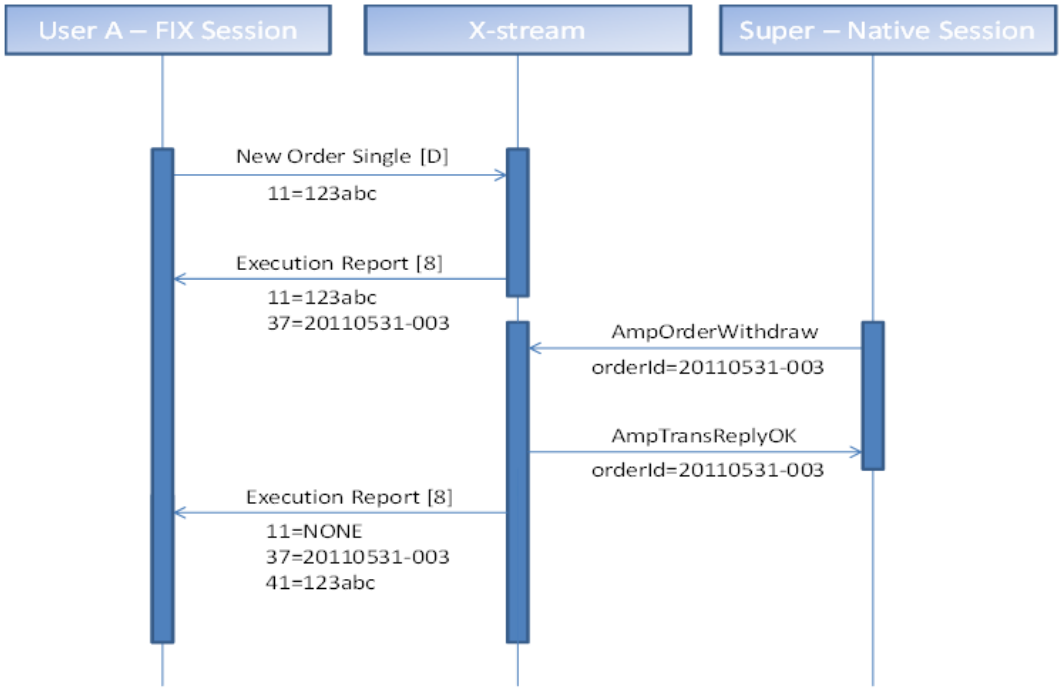
Figure 2 – Order Entered Via Native Protocol and Amended via FIX



3.4.3 Supervisor Cancellation of Orders

A supervisor may cancel orders if required. As with FIX order modified via the native protocol, the FIX client should be able to handle unsolicited messages.

Figure 3 – FIX Order Cancelled by Supervisor



3.5 On-Behalf Order Management

FIX sessions are mapped directly one-to-one to X-stream native TCP/IP sessions.

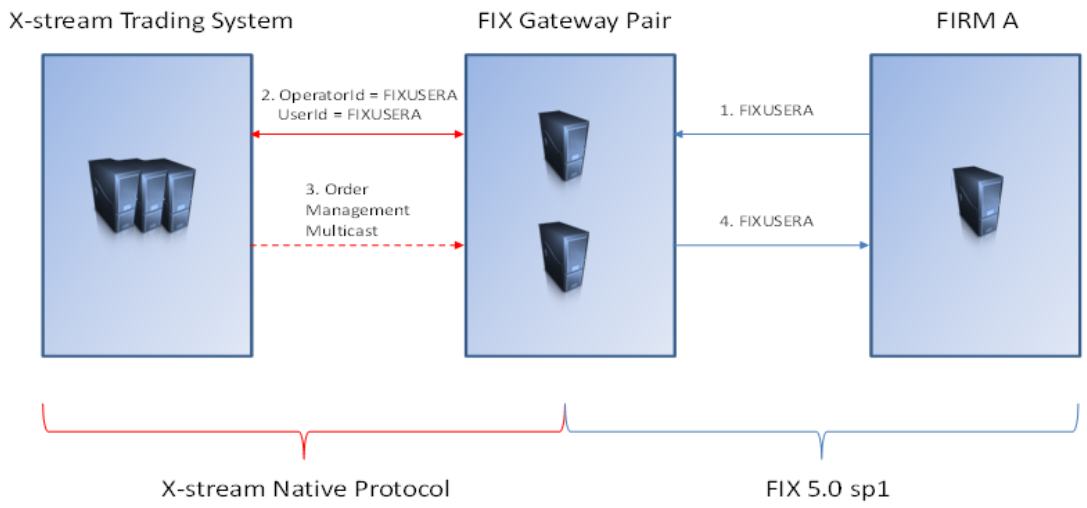
Once a FIX connection is established, the Logon (A) message initiates a native session with the X-stream trading environment.

The Username (553) and Password (554) are used to authenticate with the X-stream back-end. This Username (553) will then be used as the operator identifier for this session to X-stream.

The physical FIX session may be used for Order Management in two ways:

- The FIX userId is both operator and user for the transaction.
- The FIX userId operates 'on-behalf of' the user given in SenderSubID (50).

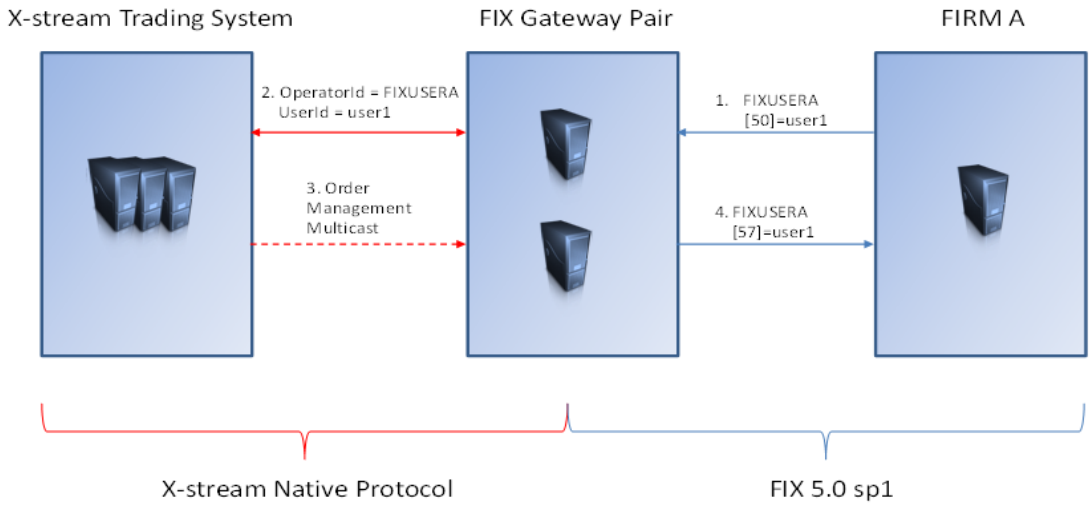
Figure 4 – Message Flow for FIX without SenderSubID (50)



A FIX session will map one-to-one with a native X-stream session. A FIX user may send order management messages without a SenderSubID (50).

The FIX gateway will send the transaction, to the back-end, with the just the Username (553) value from the originating Logon (A) message. The transaction will then be validated in X-stream using the permissions associated with this user.

Figure 5 – Message Flow where OperatorId is acting 'On-Behalf Of' UserId



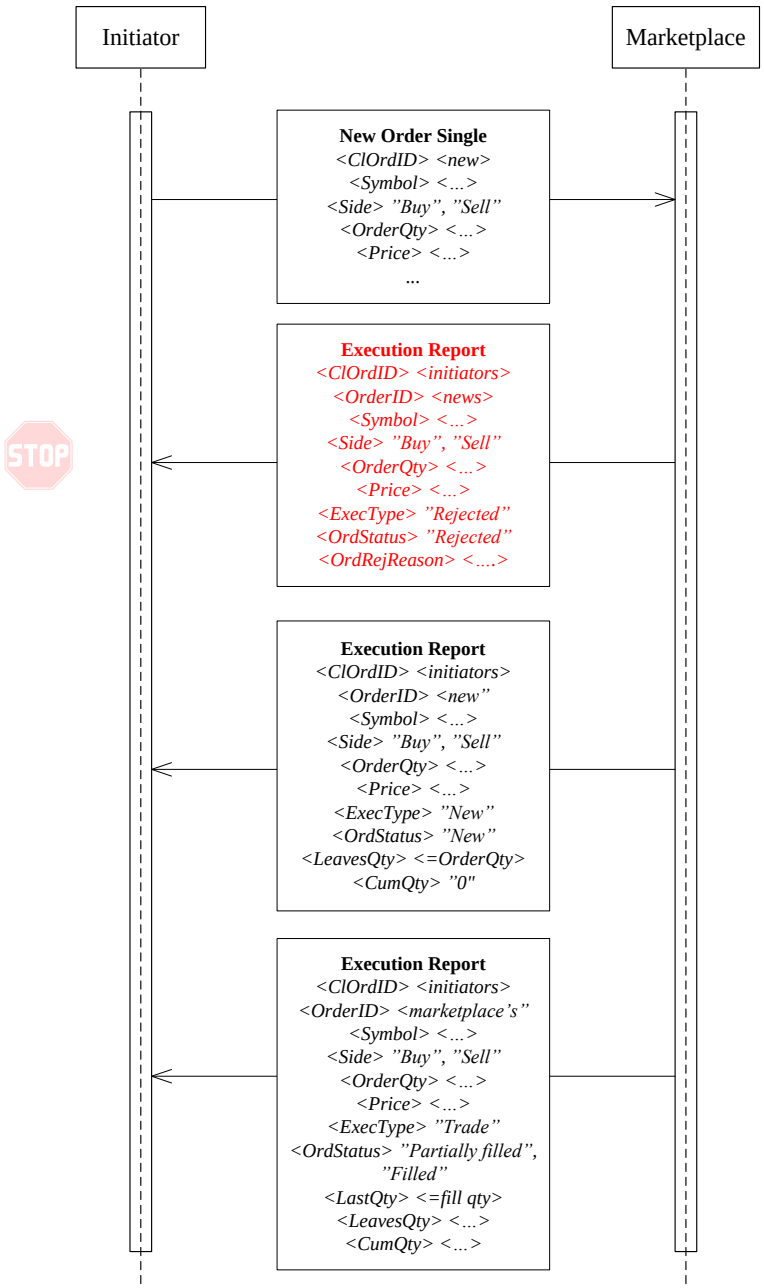
A FIX order message with SenderSubID (50) will send two usernames to the X-stream backend – OperatorId and UserId. X-stream first checks that the OperatorId, the owner of the FIX session, has permission to enter messages 'on-behalf' of the UserId from the SenderSubID (50). The transaction is then processed with the permissions of UserId.

The Execution Report (8) will contain a TargetSubID (57) that matched the inbound SenderSubID(50).

3.6 Workflows

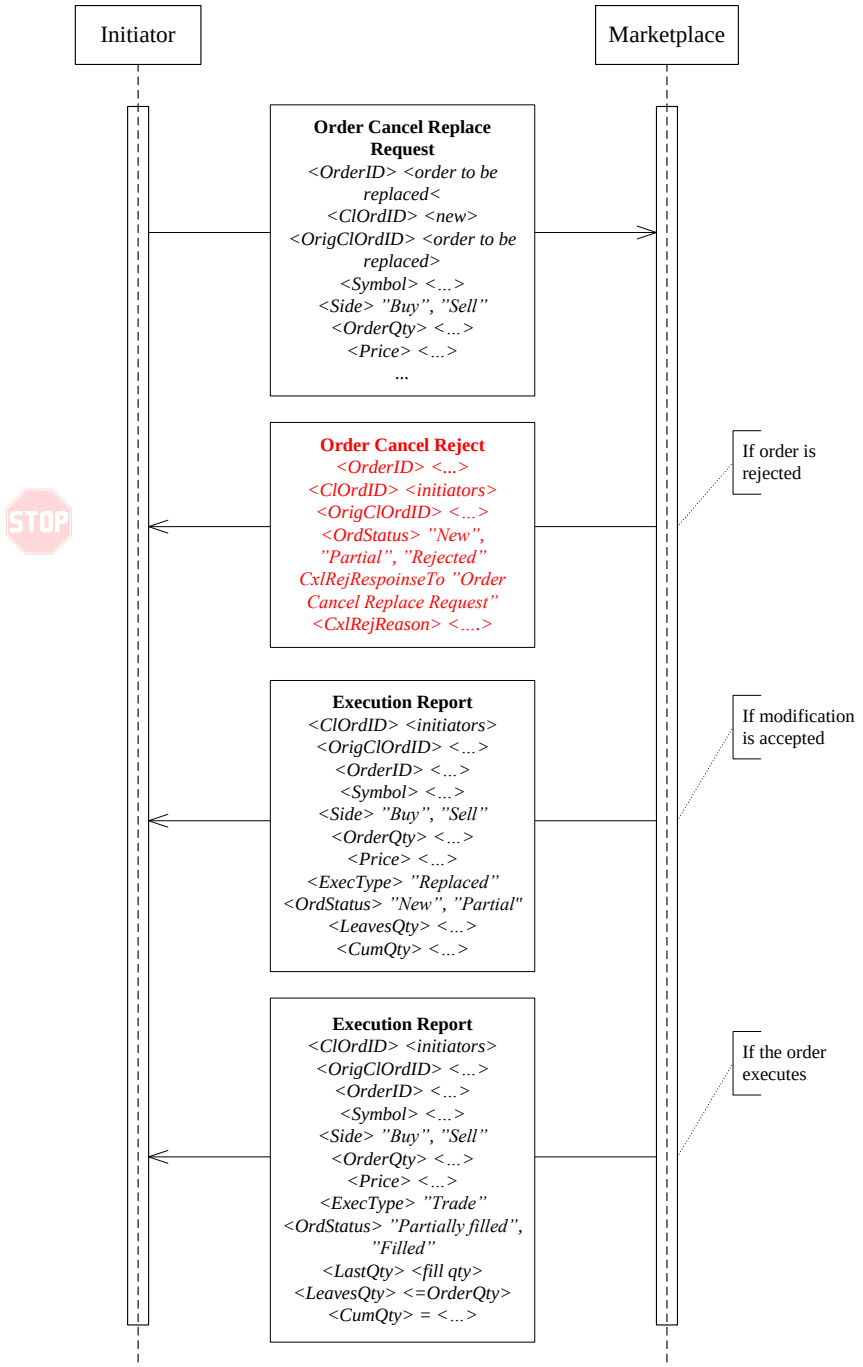
3.6.1 Entering of an New Order

Figure 6 – New Order Entry Workflow



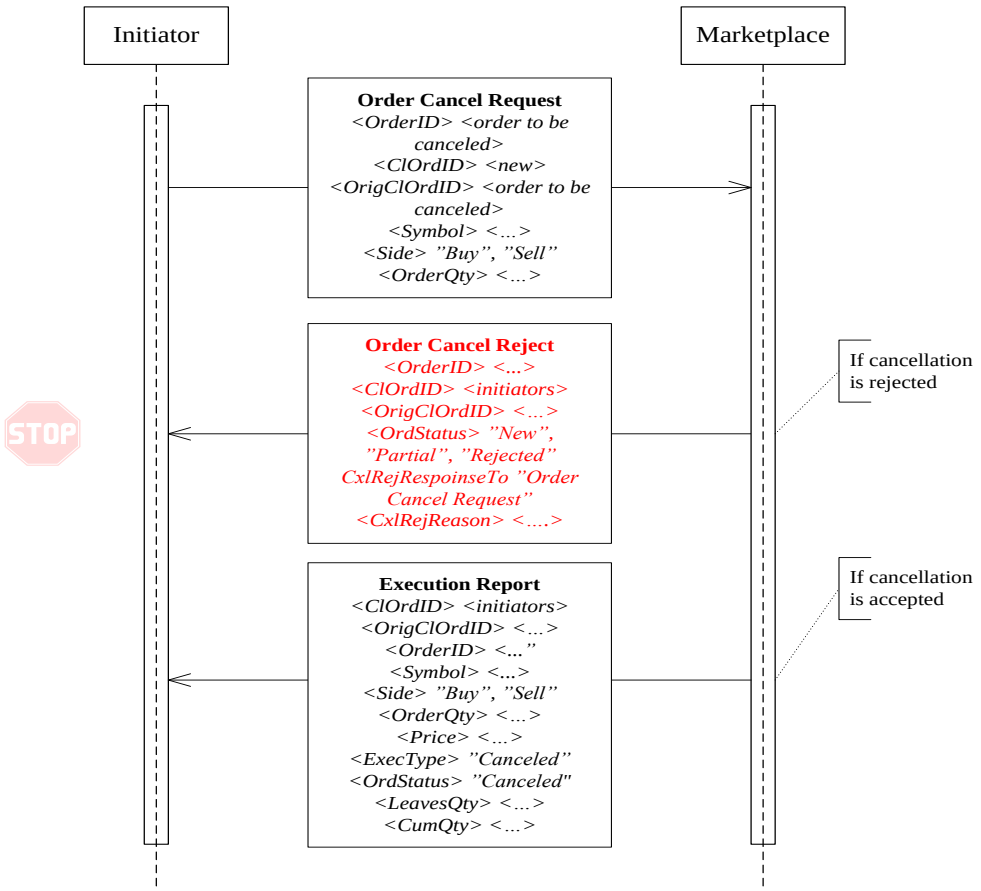
3.6.2 Modification of an Order

Figure 7 – Order Modification Workflow



3.6.3 Order Cancellation

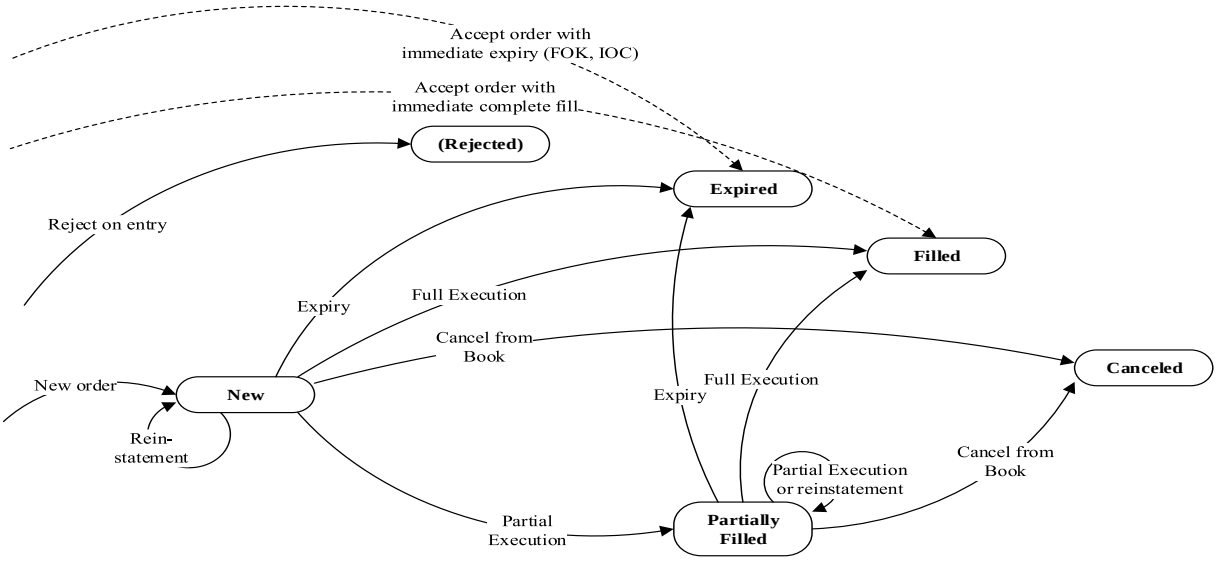
Figure 8 – Order Cancellation Workflow



3.6.4 Order Status

Order state changes are divulged in Execution Report messages. Every state change is communicated in a separate Execution Report. The OrdStatus (39) field specifies the state.

Figure 9 – Order Status States



3.7 New Order Single (D)

The new order message type is used by institutions wishing to electronically submit securities orders for execution.

Table 11 – New Order Single

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = D	
11	ClOrdID	Y	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID field.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields.	
1	Account	N	Specifies Investor Account field.	String
18	ExecInst	N	Instructions for order handling. Note for an AON or WON order, the Minimum Quantity field must not be set.	Char
38	OrderQty	Y	Quantity ordered. This value represents the number of shares for equities or par, face or nominal value	Qty

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			for Fixed Income instruments.	
40	OrdType	Y	Indicates the type of order.	Char
44	Price	Y/N	Required for all limit order types – not required for Market orders, or yield based products. For yield based products, this is the dirty price and includes accrued interest. Price (44) and Yield (236) are mutually exclusive.	Price
54	Side	Y	Side of the market (buy or sell).	Char
60	TransactTime	Y	Time of order creation by Trader. This field is not processed by the Exchange nor is it used as a mechanism to place an order at a future time.	UTCTimeStamp
110	MinQty	Y/N	Specifies the minimum fill quantity. Required if an All or None quantity condition and must be equal to the total quantity.	Qty
236	Yield	Y/N	Yield percentage – This is the yield equivalent of the price. Price (44) and Yield (236) are mutually exclusive.	Percentage
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Absence of this field indicates a 'day' order.	Char
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
126	ExpireTime	Y/N	Conditionally required if TimeInForce = 'Good till Date/Time'.	UTCTimeStamp
1138	DisplayQty	N	Replaces 'MaxFloor' and specifies the disclosed volume on hidden/iceberg orders. This is a V5.0 tag value.	Qty
513	RegistID	N	For Futures and Forwards, reference for the open trade that this order offsets.	String
58	Text	N	Use this to specify Investor Sub-account.	String
Standard Trailer		Y		

3.8 Order Cancel Request (F)

The order cancel request message requests the cancellation of **all** of the remaining quantity of an existing order. Note that the Order Cancel/Replace Request should be used to partially cancel (reduce) an order. The request will only be accepted if the order can successfully be withdrawn from the Exchange without executing.

A cancel request is assigned a ClOrdID and is treated as a separate entity. If rejected, the ClOrdID of the cancel request will be sent in the Cancel Reject message, as well as the ClOrdID of the actual order in the OrigClOrdID field. The ClOrdID assigned to the cancel request must be unique amongst the ClOrdID assigned to regular orders and replacement orders.

The format of the cancel request message is:

Table 12 – Order Cancel Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = F	
11	ClOrdID	Y	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). This identifier represents the unique identifier for the Order Cancel Request. Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID field.	String
37	OrderID	N	Unique order identifier as assigned by X-stream that identifies the Order to be changed.	String
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
38	OrderQty	Y	Order quantity	Qty
54	Side	Y	Side of the market.	Char
60	TransactTime	Y	Time this order request was initiated. This field is not processed by the Exchange nor is it used as a mechanism to cancel an order at a future time.	UTCTimeStamp
Standard Trailer		Y		

3.9 Order Cancel/Replace Request (G)

The order cancel/replace request is used to change the parameters of an existing order.

Do not use this message to cancel the remaining quantity of an outstanding order, use the Order Cancel Request message for this purpose.

Cancel/Replace will be used to change any valid attribute of an open order (i.e. reduce/increase quantity, change limit price, change instructions, etc.).

Table 13 – Order Cancel/Replace Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = G	

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
11	ClOrdID	Y	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day. Note that this identifier will be used in ClOrdID field of the Cancel Reject message if the replacement request is rejected.	String
37	OrderID	N	Unique identifier of most recent order as assigned by the Exchange.	String
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields. Must match original order	
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields.	
1	Account	N	Specifies Investor Account.	String
18	ExecInst	N	Instructions for order handling. Note for an AON or WON order, the Minimum Quantity field must not be set.	Char
38	OrderQty	Y	Quantity ordered. This value represents the number of shares for equities or par, face or nominal value for Fixed Income instruments.	Qty
40	OrdType	Y	Indicates the type of order to change to (must follow rules of the Exchange).	Char
44	Price	Y/N	Required for all limit order types, but not for yield based products. For yield based products, this is the dirty price and includes accrued interest. Price (44) and Yield (236) are mutually exclusive.	Price
54	Side	Y	Side of the market.	Char
60	TransactTime	Y	Time of execution/order creation. This field is not processed by the Exchange nor is it used as a mechanism to amend an order at a future time.	UTCTimeStamp
110	MinQty	N	Specifies the minimum fill quantity	Qty

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment.	Char
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
126	ExpireTime	Y/N	Conditionally required if TimeInForce = 'Good till Date/Time'.	UTCTimeStamp
236	Yield	N	Yield percentage.	Percentage
513	RegistID	N	For Futures and forwards, reference for the open trade that this order offsets.	String
1138	DisplayQty	N	Replaces 'MaxFloor' and specifies the disclosed volume on hidden/iceberg orders. This is a V5.0 tag value.	Qty
58	Text	N	Used to specify Investor Subaccount.	String
Standard Trailer		Y		

3.10 Order Cancel Reject (9)

The order cancel reject message is issued by the Exchange upon receipt of a cancel request or cancel/replace request message which cannot be honoured. Filled orders cannot be changed.

When rejecting a Cancel/Replace Request (or Cancel Request), the Cancel Reject message should provide the ClOrdID which was specified on the Cancel/Replace Request (or Cancel Request) message for identification, and the OrigClOrdId should be that of the last accepted order except in the case of CxlRejReason = "Other".

Refer to the Text (58) field for specific information on the reason for the rejection.

When rejecting an Order Mass Action Request specifying Order Cancellation, the ClOrdID should be set to the ClOrdID value of the Order Mass Action Request. OrigClOrdID is not specified for a rejected Order Mass Action Requests.

The order cancel reject message format is as follows.

Table 14 – Order Cancel Reject

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = 9	
11	ClOrdID	Y	Unique identifier for Order as assigned by sell-side (e.g. exchange, ECN). If CxlRejReason="Unknown order" specify "NONE".	String
37	OrderID	Y	Unique identifier of most recent order as assigned by the Exchange. If CxlRejReason="Unknown order", specify "NONE".	String
39	OrdStatus	Y	Describes the current status of the order	Char
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
60	TransactTime	Y	Time of order cancellation request rejection by the Exchange.	UTCTimeStamp
102	CxlRejReason	Y	Code to identify reason for cancel rejection. Only '99' (Other) will be returned. Refer to 'text' (58) for exact reason for rejection.	Int
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to.	Char
58	Text	N	Free format text string	String
Standard Trailer		Y		

3.11 Order Status Request (H)

The order status request message is used by the broker/participant to generate an order status message back from the Exchange.

If an Order Status Request is issued for an order that is either cancelled, expired or fully filled, only mandatory fields will be provided in resulting Execution Reports. Non-mandatory fields will not be provided.

Table 15 – Order Status Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = H	
11	ClOrdID	Y	Corresponds to the ClOrdID (11) of the order whose status is being requested (if it exists). Conditionally required if the OrderID(37) is not provided. Either OrderID or ClOrdID must be provided.	String
37	OrderID	N	Conditionally required if ClOrdID(11) is not provided. Either OrderID (37) or ClOrdID (11) must be provided.	String
790	OrdStatusReqID	N	Optional, can be used to uniquely identify a specific Order Status Request message. Echoed back on Execution Report if provided.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
54	Side	Y	Side of the market.	Char
StandardTrailer		Y		

3.12 Order Mass Action Request (CA)

The Order Mass Action Request message can be used to request the cancellation or status of a group of orders that match the criteria specified within the request.

Table 16 - Order Mass Action Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = CA	
11	ClOrdID	Y	Unique ID of Order Mass Action Request as assigned by the institution. Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID field.	String
526	SecondaryClOrdID	N	Assigned by order originator.	String
584	MassStatusReqID	Y	Value assigned by issuer of Order Mass Action Request to uniquely identify the request. This ID will be returned on the Execution report.	String
1373	MassActionType	Y	Specifies the type of action requested. This is a V5.0 tag value.	Int
1374	MassActionScope	Y	Specifies the scope of the action. This is a V5.0 tag value.	Int
1301	MarketID	N	MarketID for which orders are to be affected. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security trades. It is mapped to X-stream Board Id. This is a V5.0 tag value.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		N	Insert here the set of "Instrument" (symbology) fields.	
54	Side	N	Side of the market.	Char
60	TransactTime	Y	Time of mass order action request by Trader. This field is not processed by the Exchange nor is it used to schedule an action at a future time.	UTCTimeStamp
58	Text	N	Free format text string	String
Standard Trailer		Y		

3.13 Order Mass Action Report (BZ)

The Order Mass Action Report is used to acknowledge an Order Mass Action Request. Note that each order that is affected by the Order Mass Action Request is acknowledged with a separate Execution Report for each order.

Table 17 - Order Mass Action Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BZ	
11	ClOrdID	N	ClOrdID provided on the Order Mass Action Request.	String
526	SecondaryClOrdID	N	Assigned by order originator.	String
1369	MassActionReportID	Y	Unique Identifier for the Order Mass Action Report. This is a V5.0 tag value.	String
1373	MassActionType	Y	Specifies the type of mass action requested. This is a V5.0 tag value.	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. This is a V5.0 tag value.	Int
1375	MassActionResponse	Y	Indicates the action taken by the counterparty order handling system as a result of the Action Request. This is a V5.0 tag value.	Int
1376	MassActionRejectReason	N	Indicates why Order Mass Action Request was rejected. Required if MassActionResponse = 0 Reason Order Mass Action Request was rejected. This is a V5.0 tag value.	Int
533	TotalAffectedOrders	N	Optional field used to indicate the total number of orders affected by the Order Mass Action Request	Int
Start of Component block, expanded in line < AffectedOrdGrp >				
534	NoAffectedOrders	N	Optional field used to indicate the number of order identifiers for orders affected by the Order Mass Action Request. Must be followed with OrigClOrdID as the next field	Int
→ 41	OrigClOrdID	N	Required if NoAffectedOrders > 0 and must be the first repeating field in the group. Indicates the client order id of an order affected by this request. If order(s) were manually delivered (or otherwise not delivered over FIX and not assigned a ClOrdID) this field should contain string "MANUAL".	String
End of Component block, expanded in line < AffectedOrdGrp >				
1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security trades. It is mapped to X-stream Board Id. This is a V5.0 tag value.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm	

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			identification) fields.	
Component block <Instrument>		N	Insert here the set of "Instrument" (symbology) fields.	
54	Side	N	Side of the market.	Char
60	TransactTime	N	Equal to time of Order Mass Action Request.	UTCTimeStamp
58	Text	N	Free format text string	String
Standard Trailer		Y		

3.14 Execution Report (8)

The execution report message is used to:

1. Confirm the receipt of an order
2. Confirm changes to an existing order (i.e. accept cancel and replace requests)
3. Report order status information
4. Report fill information on working orders
5. Report fill information on tradeable or restricted tradeable quotes
6. Report on rejected order
7. Report on orders activated/deactivated by Market Control
8. Report on orders with triggers that have been activated. Refer to Appendix B.4 for additional details on Triggered Orders.

Table 19, entitled 'Execution Report Returned Tags Based On Scenario' follows the Execution Report message description and provides information on which tags are returned in an Execution Report message based on various order management scenarios.

If an Order Status Request is issued for an order with an OrdStatus(39) of either Cancelled, Expired or Filled, only mandatory fields will be provided in resulting Execution Reports. Non-mandatory fields will not be provided.

Table 18 – Execution Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = 8	
11	ClOrdID	Y/N	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			field. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID(11). In the case of quotes can be mapped to the QuoteID (117) tag of a Mass Quote.	
17	ExecID	Y	Unique identifier of execution message as assigned by the Exchange (will be 0 (zero) for ExecType=I (Order Status)).	String
18	ExecInst	N	Instructions for order handling.	Char
37	OrderID	Y	OrderID is required to be unique for each chain of orders.	String
41	OrigClOrdID	Y/N	Conditionally required for response to a Cancel or Cancel/Replace request	String
150	ExecType	Y	Type of Execution being reported. Describes the specific ExecutionRpt (i.e. Pending Cancel) while OrdStatus (39) will always identify the current order status (i.e. Partially Filled).	Char
526	SecondaryClOrdID	N	Assigned by the party that originates the order.	String
584	MassStatusReqID	Y/N	Required if responding to an Order Mass Status Request. Echos back the value provided by the requester.	String
790	OrdStatusReqID	N	Required if responding to and if provided on the Order Status Request message. Echo back the value provided by the requester.	String
911	TotNumReports	N	Can be used when responding to an Order Mass Status Request to identify the total number of Execution Reports which will be returned. Not Supported.	Int
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields.	
1	Account	N	Specifies Investor Account.	String
6	AvgPx	N	Calculated average price for all fills on this order during the day. If not available then the value reflects the trade price for this fill.	Price
14	CumQty	Y	Total matched quantity.	Qty
31	LastPx	N	Price of this fill.	Price

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
32	LastQty	N	Quantity (e.g. shares) bought/sold on this fill.	Qty
38	OrderQty	N	Quantity ordered.	Qty
110	MinQty	N	Minimum fill quantity.	Qty
39	OrdStatus	Y	Describes the current state of an order.	Char
40	OrdType	N	OrderType	Char
44	Price	N	Price on order.	Price
54	Side	Y	Side of order.	Char
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Absence of this field indicates a 'day' order.	Char
60	TransactTime	Y	Time of execution/order creation (expressed in Universal Time Coordinated (UTC), also known as GMT.	UTCTimeStamp
75	TradeDate	N	Indicates date of trade referenced in this message in YYYYMMDD format.	LocalMktDate
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
126	ExpireTime	Y/N	Conditionally required if TimeInForce = GTD and ExpireDate is not specified.	UTCTimestamp
64	SettlDate	N	Specific date of trade settlement Settlement Date is in YYYYMMDD format.	LocalMktDate
103	OrdRejReason	N	For optional use with ExecType = 8 (Rejected). Code to identify reason for order rejection.	Int
151	LeavesQty	Y	Quantities open for further execution. If the OrdStatus is Cancelled, DoneForTheDay, Expired or Rejected (in which case the order is no longer active) then LeavesQty could be 0, otherwise LeavesQty = OrderQty – CumQty.	Qty
236	Yield	N	Yield percentage (Fixed Income only)	Percentage
381	GrossTradeAmt	N	Total amount traded expressed in units of currency. Calculated on Price*LastQty	Amt
880	TradeMatchID	N	Identifier assigned by the trading system for a trade. This is the X-stream trade id.	String
1057	AggressorIndicator	N	Used to identify whether the order initiator is an aggressor or not in the trade. Valid during continuous trading only. This is a V5.0 tag value.	Boolean

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
1138	DisplayQty	N	Replaces 'MaxFloor' and specifies the disclosed volume on hidden/iceberg orders. This is a V5.0 tag value. This field is always returned as part of a fill or partial fill for all order types. For non-hidden/iceberg orders this field will contain the same value as LeavesQty (151).	Qty
513	RegistID	N	For Futures and forwards, reference of the open trade that this order offsets.	String
58	Text	N	On an error condition, this will specify X-stream generated error message.	String
797	CopyMsgIndicator	N	Drop Copy	Boolean
Standard Trailer		Y		

Table 19 – Execution Report Returned Tags Based On Scenario

	ClOrdID (11)	ExecID (17)	ExecInst (18)	OrderID (37)	OrigClOrdID (41)	ExecType (150)	SecondaryClOrdID (526)	MassStatusReqID (584)	TotNumReports (911)	OrdStatusReqID (790)	Parties	Instrument	Trigger Instruction	Account (1)	AvgPX (6)	CumQTY (14)	LastPX (31)	LastQTY (32)	MinQty (110)	OrderQty (38)	OrdStatus (39)	OrdType (40)	Price (44)	Side (54)	TimeInForce (59)	TransactTime (60)	TradeDate (75)	DisplayQty (1138)	ExpireDate (432)	ExpireTime (126)	SettleDate (64)	SecuritySubType (762)	CxlRejReason (102)	CxlRejReaResponsetTo (434)	OrdRejReason (103)	LeavesQTY (151)	Yield (236)	GrossTradeAmt (381)	TradeMatchID (880)	AggressorIndicator (1057)	Text (58)
New Order Single	R	R	C	R		R	C				R	R	C	C		R	C	C	C	R	R	R	R	R	C	R		C	C	C		C				R	C				C
Order Cancel Pending	C	R		R	R	R						R				R				R	R			R	C			C								R					C
Order Cancel / Replace	C	R	C	R	R	R		C	C		R	R	C	C		R	C	C	C		R		R	R	C	R		C	C	C		C				R	C				C
Order Cancelled	C	R		R	R	R		C	C			R				R	C	C		R	R	C	R	R	C	R		C				C				R					C
Order Filled	R	R	C	R		R					R	R	C	C	R	R	R	R	C	R	R	C	R	R	C	R		C			R	R				R	C	R	R	R	C
Order Partially Filled	R	R	C	R		R					R	R	C	C	R	R	R	R	C	R	R	C	R	R	C	R		C	C	C	R	R				R	C	R	R	R	C
Order Rejected	C	R		R		R						R				R				C	R			R	C	R		C	C	C					R						C
Order Status	R	R	C	R	C	R	C	C	C	C	R	R	C	C		R			C	R	R	R	R	R	C	R		C	C	C		C				R	C				C

C = Conditional - Based on input transaction/query (or error condition)

R = Returned as part of Execution Report message

4 Quote Management

Market makers can submit quotes through NSE FIX. The quote management category consists of the following messages:

- Mass Quote
- Mass Quote Acknowledgement.

4.1 Unique QuoteID (117)

X-stream will not check for uniqueness of QuoteID (117) on MassQuote message. Firms submitting order transactions via FIX interface must ensure unique QuoteID (117) is entered on these transactions.

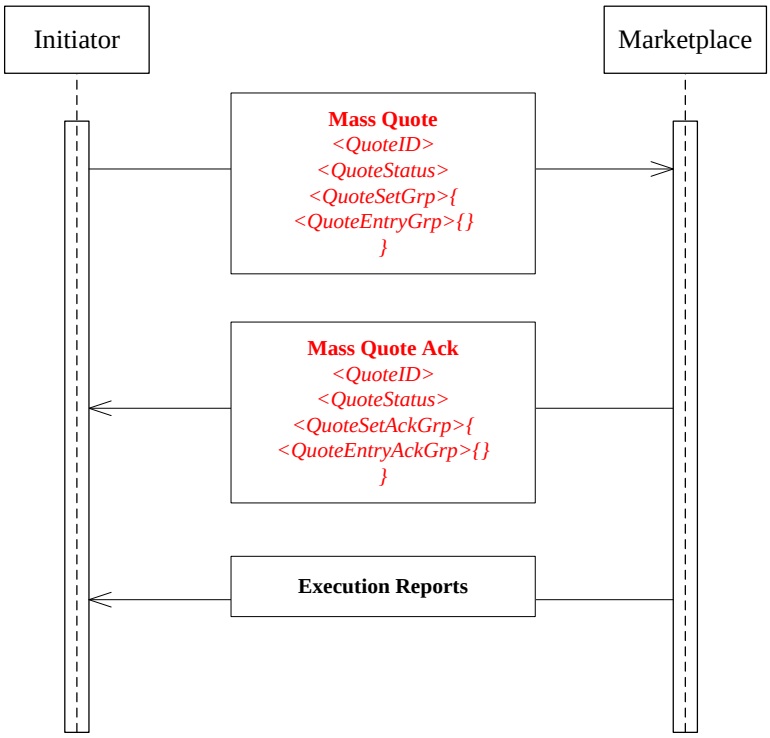
4.2 Workflows

4.2.1 Entering of a mass quote

Mass Quote contains a group of Quote Sets. A Quote Set contains a group of Quote Entries. Each quote entry can submit a pair of bid and offer quotes on a single instrument. Existing quotes in the instrument will be replaced.

Mass Quote Acknowledgement is used as the application level response to a Mass Quote message. It reports the status of a Mass Quote, as well as the result of every Quote Set and Quote Entry.

Figure 10 – Mass Quote workflow



Execution Reports will be generated for quotes individually. The ClOrdID(11) field will be set to QuoteID(117), the SecondaryClOrdID(526) will be set to QuoteEntryID(299). Every quote will

be assigned an unique OrderID(37), which can be used in Order Cancel Replace Request(G) to amend the quote. However OrigClOrdID(41) can not be used to amend a quote.

4.2.2 Quote Entry Cancel

A quote can be cancelled by sending a quote entry with bid or offer prices and sizes all set to zero in a Mass Quote message. In this case the Mass Quote Acknowledgement will indicate that the quote is cancelled. If a quote is cancelled by the system or expired, an ExceptionReport will be generated.

4.3 Mass Quote (i)

The Mass Quote message can contain quotes for multiple securities to support applications that allow for the mass quoting of an option series. Two levels of repeating groups have been provided to minimize the amount of data required to submit a set of quotes for a class of options (e.g. all option series for IBM).

NSE FIX supports only one QuoteSetGrp per Mass Quote message. Fragmented Mass Quote messages will be rejected - NoQuoteEntries(295) should always be the same as TotNoQuoteEntries(304) in the same QuotSetGrp.

Table 20 – Mass Quote

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = i	
117	QuoteID	Y	Unique identifier for quote	String
1	Account	N	Specifies trade account type.	String
Start of Component block, expanded in line < QuotSetGrp >				
296	NoQuoteSets	Y	The number of sets of quotes in the message	NumInGrp
→	302 QuoteSetID	Y	Sequential number for the Quote Set. For a given QuoteID – assumed to start at 1. Must be the first field in the repeating group.	String
→	367 QuoteSetValidUntilTime	N	Indicates expiration time of this particular QuoteSet (always expressed in UTC (Universal Time Coordinated, also known as "GMT")	UTCTimestamp
→	304 TotNoQuoteEntries	Y	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set.	Int
→	Start of Component block, expanded in line < QuotEntryGrp >			
→	295 NoQuoteEntries	Y	The number of quotes for this QuotSet that follow in this message.	NumInGrp
→	→ 299 QuoteEntryID	Y	Uniquely identifies the quote across the complete set of all quotes for a given quote provider.	String

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
→	→	Component block <Instrument>	Y	Insert here the set of "Instrument" (symbology) fields defined in "Common Components of Application Messages.	
→	→	132 BidPx	N	Bid price/rate.	Price
→	→	133 OfferPx	N	Offer price/rate.	Price
→	→	134 BidSize	N	Quantity of bid	Qty
→	→	135 OfferSize	N	Quantity of offer	Qty
→	→	62 ValidUtilTime	N	Indicates expiration time of quote (always expressed in UTC (Universal Time Coordinated, also known as "GMT")	UTCTimestamp
→	End of Component block, expanded in line < QuotEntryGrp >				
End of Component block, expanded in line < QuotSetGrp >					
Standard Trailer			Y		

4.4 Mass Quote Acknowledgement (b)

Mass Quote Acknowledgement is used as the application level response to a Mass Quote message. The Mass Quote Acknowledgement contains a field for reporting the reason in the event that the entire quote is rejected (QuoteRejectReason[300]). The Mass Quote Acknowledgement also contains a field for each quote that is used in the event that the quote entry is rejected (QuoteEntryRejectReason[368]).

Table 21 – Mass Quote Acknowledgement

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = b	
117	QuoteID		Y	Unique identifier for quote	String
297	QuoteStatus		N	Status of the mass quote acknowledgement.	Int
300	QuoteRejectReason		N	Reason Quote was rejected.	Int
1	Account		N	Specifies trade account type.	String
58	Text		N	Free format text string	String
Start of Component block, expanded in line < QuotSetAckGrp >					
296	NoQuoteSets		Y	The number of sets of quotes in the message	NumInGrp
→	302	QuoteSetID	Y	Sequential number for the Quote Set. For a given QuoteID - assumed to start at 1. Must be the first field in the repeating group.	String

TAG	FIELDNAME			REQ'D	COMMENTS	FORMAT
→	304	TotNoQuoteEntries		N	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set. Required if NoQuoteEntries > 0	Int
→	893	LastFragment		N	Indicates whether this is the last fragment in a sequence of message fragments. Only required where message has been fragmented.	Boolean
→	Start of Component block, expanded in line < QuotEntryAckGrp >					
→	295	NoQuoteEntries		Y	The number of quotes for this QuotSetAck that follow in this message.	NumInGrp
→	→	299	QuoteEntryID	Y	Uniquely identifies the quote across the complete set of all quotes for a given quote provider.	String
→	→	Component block <Instrument>		Y	Instrument component received in QuotEntryGrp	
→	→	132	BidPx	N	Bid price/rate.	Price
→	→	133	OfferPx	N	Offer price/rate.	Price
→	→	134	BidSize	N	Quantity of bid	Qty
→	→	135	OfferSize	N	Quantity of offer	Qty
→	→	62	ValidUtilTime	N	Indicates expiration time of quote (always expressed in UTC (Universal Time Coordinated, also known as "GMT")	UTCTimestamp
→	→	1167	QuoteEntryStatus	N	Identifies the status of an individual quote.	Int
→	→	368	QuoteEntryRejectReason	N	Reason Quote Entry was rejected.	Int
→	End of Component block, expanded in line < QuotEntryAckGrp >					
End of Component block, expanded in line < QuotSetAckGrp >						
Standard Trailer				Y		

5 Trade Capture Reporting

Trade Capture reports are used for a variety of purposes and include:

- Relaying Confirmed Trades to various parties not directly involved in the execution, such as CSDs, clearing houses, clearing firms and regulatory bodies. Those messages are outbound from the marketplace.
- Relaying Confirmed Trades to counterparties of the trade. Those messages are outbound from the marketplace.
- Reporting of privately negotiated trades. Those messages are inbound to the marketplace and may also be outbound.

5.1 Trade Capture Messages

The Trade Capture category of messages consists of the following:

- Trade Capture Report Request
- Trade Capture Report
- Trade Capture Report Request Ack
- Trade Capture Report Ack

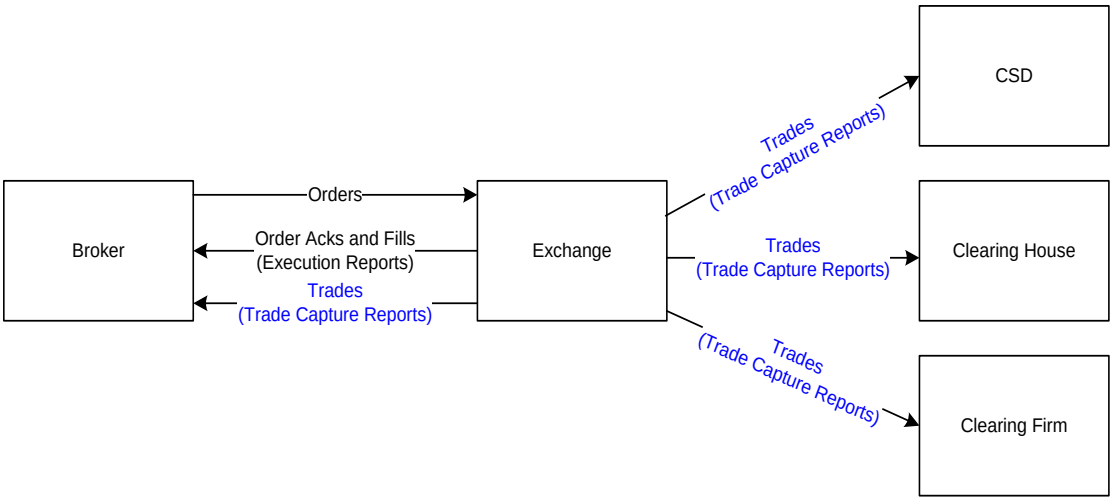
5.2 Workflows

The set of trade capture workflows consist of workflows for third party trade capture, e.g. CSD, Regulatory, ECN reporting, etc.

Reporting of privately negotiated trades, i.e. trades formed outside of the marketplace. Those messages are **inbound** (to the marketplace) and may also be **outbound** (when the marketplace relays them to counterparties). By convention the Seller always report two sided negotiated trades.

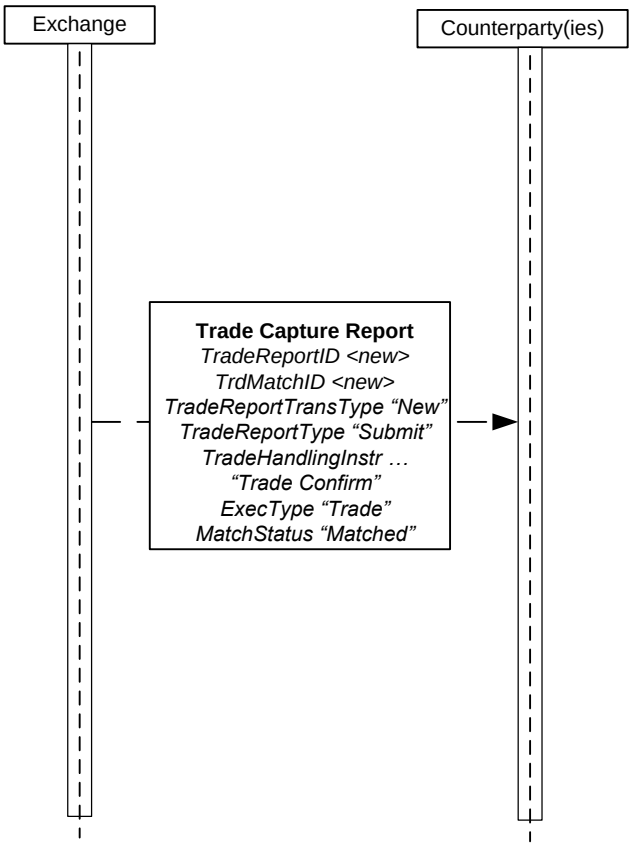
5.2.1 Trade Capture Workflow for Multiple Counterparties

Figure 11 – Trade Capture High Level Workflow



5.2.2 Workflow for Third Party Trade Capture

Figure 12 – Third Party Trade Capture

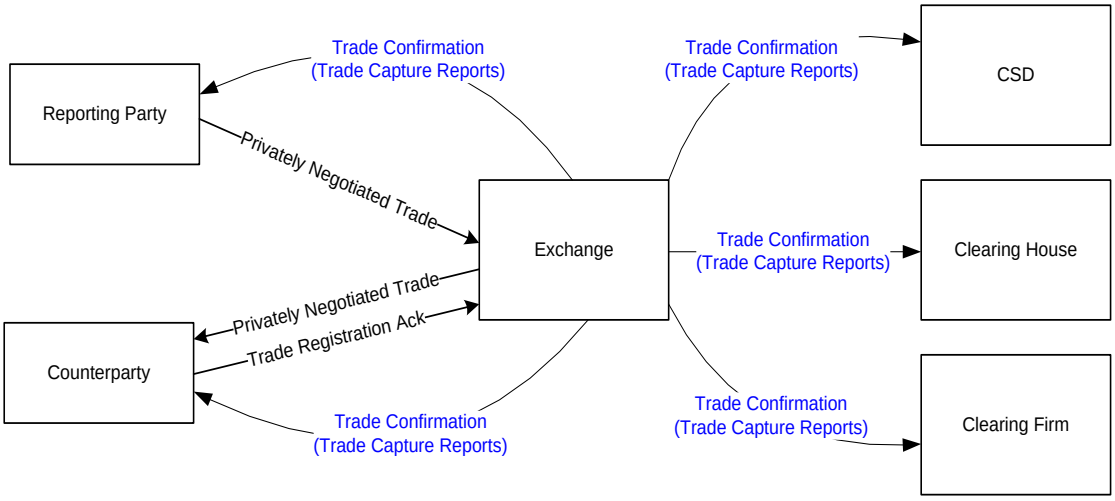


5.2.3 Trade Capture Workflow for Privately Negoiated Trades

The deal is struck between two parties, one of whom has an obligation to report the trade. The counterparty does not have agreement with the reporting party, so he must acknowledge the trade. The reporting party sends the trade report to the market. The market informs the counterparty of the report and the counterparty then accepts the trade. The marketplace confirms the Confirmed Trade to all involved parties. The FIX Trade Capture Report is used for all involved messages. Both parties must subscribe to Trade Capture Reporting to be able to receive the final Trade Confirmations.

When submitting a one Party (crossing) Trade Capture Report, the submitter must fill in details of both sides in two TrdCapRptSideGrps. The Exchange will either confirm the TradeCaptureReport (without referring to the counterparty), or reject with TradeCaptureReportAck.

Figure 13 – Trade Capture High Level Workflow – Negoiated Deal



5.2.4 Workflow for One-Party Report for Pass-through to Counterparty

1. The initiator (seller) sends TradeCaptureReport (AE) with a unique TradeReportID (571).
2. If rejected, the marketplace will send to the initiator a TradeCatureReport (AE) with the TradeReportID (571) set to the same as the TradeReportID (571) received.
3. If accepted, the marketplace will send to the initiator a TradeCatureReport (AE) with a new TradeReportID (571), a new ExecID (17) and the TradeReportRefID (572) set to the initiator's TradeReportID (571). The Marketplace will also send a TradeCatureReport (AE) to the counterparty, with a new TradeReportID (571) and a new ExecID (17).
4. The initiator can withdraw the TradeCaptureReport with a new TradeReportID (571), a ExecID (17) that is set to the same ExecID in the TradeCaptureReport received from the Marketplace in step 3. And tradeReportRefID (572) set to the same as the TradeReportID (571) received in step 3.
5. The counterparty can either accept or reject the alleged TradeCaptureReport with a new TradeCaptureReport, by setting the ExecID (17) to the same as the ExecID (17) in the TradeCaptureReport received from the Marketplace and the TradeReportRefID (572) set to the TradeReportID (571) in the TradeCaptureReport received from the Marketplace.
6. If the counterparty confirms the deal the Marketplace will send a TradeCaptureReport to both parties, with a new ExecID (17). The TradeReportType (856) will be set to '0' – submit.
7. If the counterparty delines or the intiator withdraws the trade, the Marketplace will send a TradeCaptureReport with ExecID (17) set to be the same as ExecID (17) in step 3, TradeReportID (571) will be set to the same TradeReportID (571) received in step 3 and (for the initator only) the TradeReportRefID (572) will be set to the same as the TradeReportID (571) sent in step 1.

Figure 14 - One-Party Report for Passs-through to Counterparty

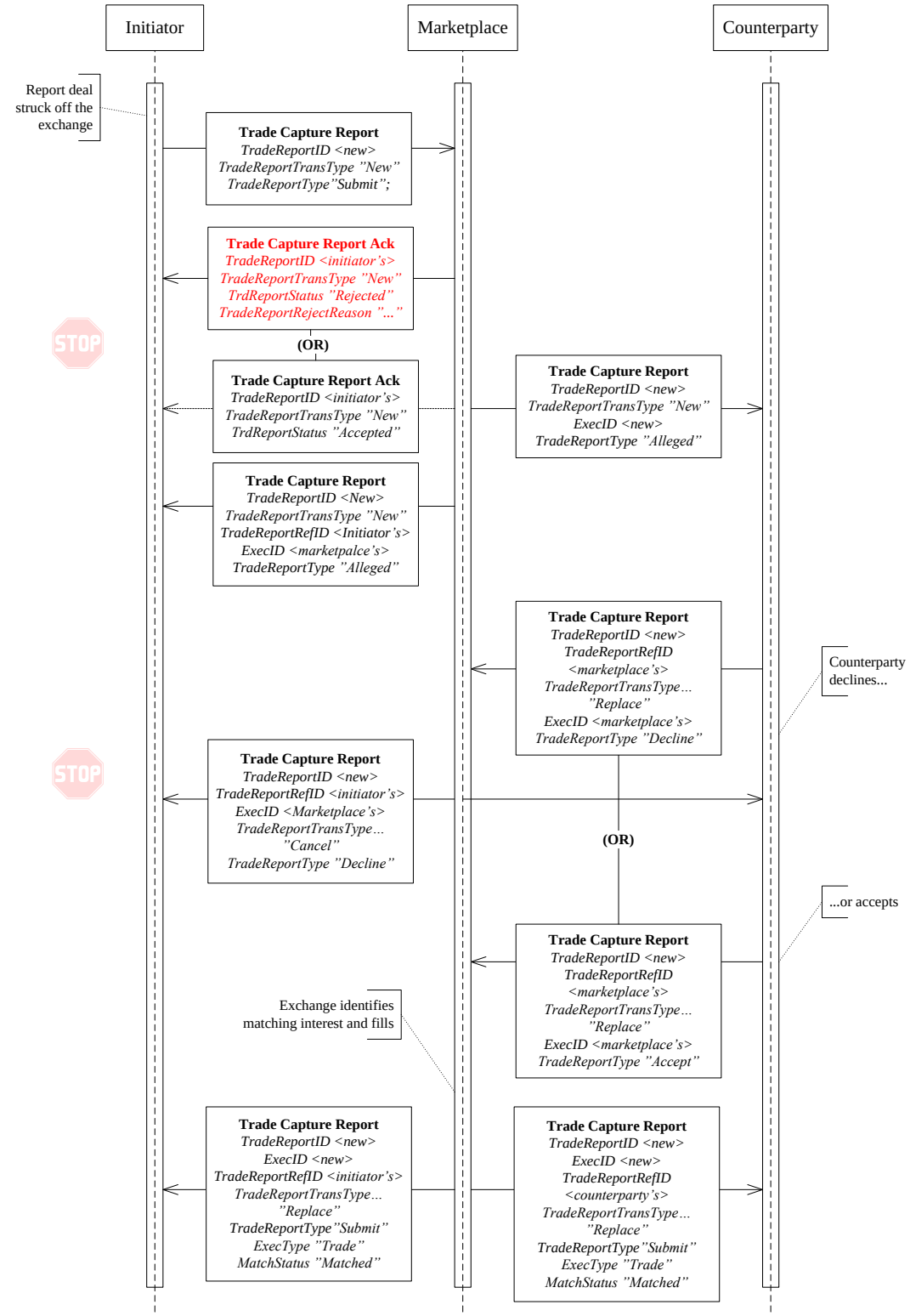
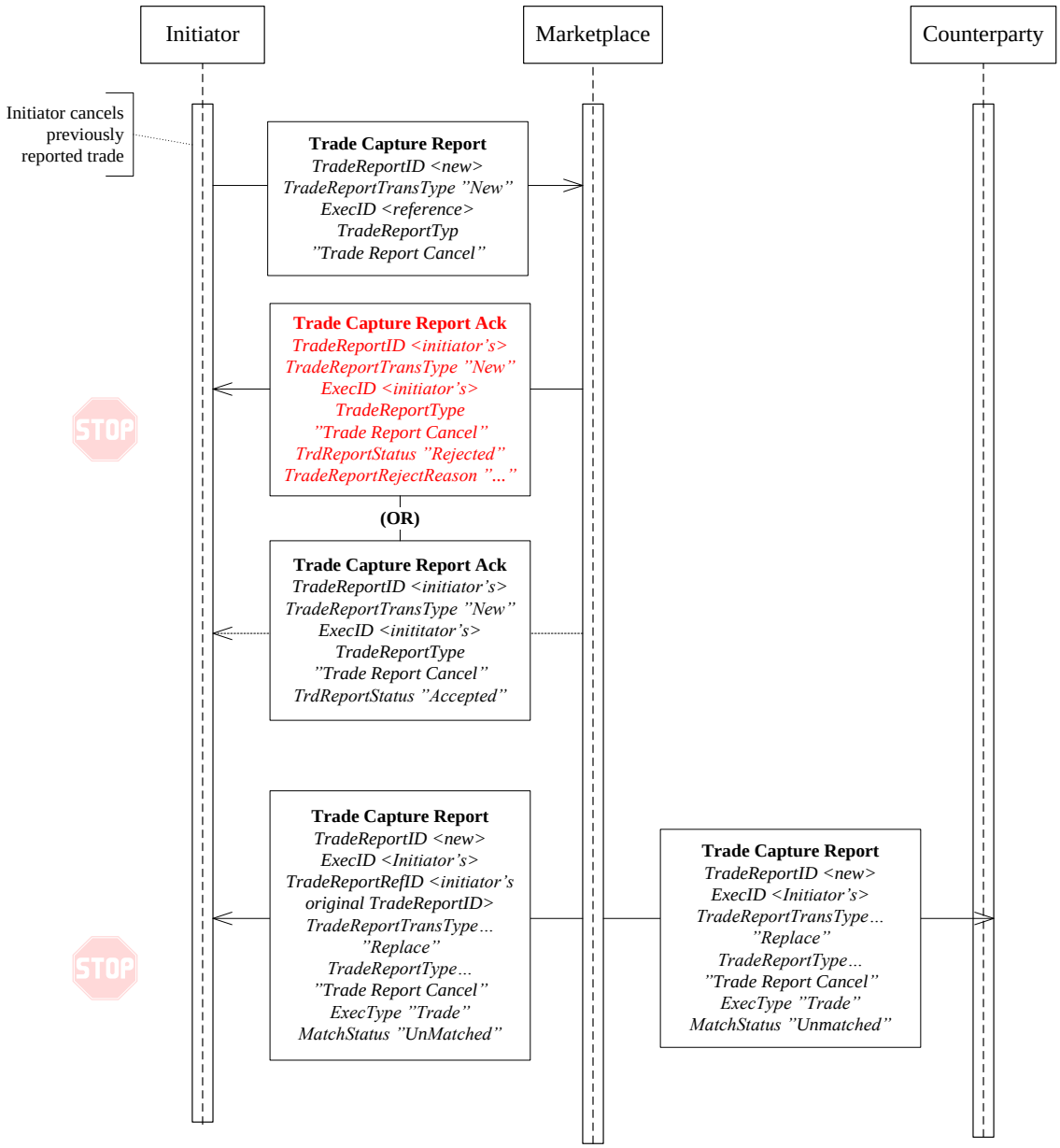


Figure 15 - One-Party cancel Report (or time out) before counterpary confirms



5.3 Trade Capture Report Request (AD)

The Trade Capture Report Request is used to:

- Request all trades that the FIX user can see.
- Subscribe or unsubscribe for trade capture reports. By default only a snapshot report will be provided (no automatic updates).

The response to the Trade Capture Report Request consists of one or more Trade Capture Reports or a Trade Capture Report Request Ack followed by one or more Trade Capture Reports.

Table 22 – Trade Capture Report Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AD	
568	TradeRequestID	Y	Identifier for the trade request	String
569	TradeRequestType	Y	Type of Trade Capture Report.	Int
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default (snapshot only – no subscription).	Char
58	Text	N	Free format text string	String
StandardTrailer		Y		

5.4 Trade Capture Report (AE)

The Trade Capture Report message can be:

- Sent as a reply to a Trade Capture Report Request.
- Can be used to report off market trades.

Table 23 – Trade Capture Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AE	
571	TradeReportID	N	Unique identifier for the Trade Capture Report	String
17	ExecID	N	Exchanged assigned Execution ID (Trade Identifier).	String
	Fields for Negotiated Deals			
487	TradeReportTransType	Y/N	Set to 0 for entering a Neg Deal	Int
856	TradeReportType	Y/N	'0' for submit, '1' Alleged, '3' Decline	Int
150	ExecType	Y/N	'F' - Trade	Char
828	TrdType	Y/N	Type of Trade - 22 (Private Negotiated Deal).	Int
572	TradeReportRefID	Y/N	Used by the Marketplace to identify corresponding Client Submission.	String
880	TradeMatchID	N	Identifier assigned by the trading system for a trade. This is the X-stream trade id.	String

1126	OrigTradeID	N	The original trade ID of the trade that has been amended.	String
150	ExecType	N	Type of Execution being reported.	Char
325	UnsolicitedIndicator	N	Set to 'Y' if message is sent as a result of a subscription request or out of band configuration as opposed to a Position Request.	Boolean
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default.	Char
568	TradeRequestID	N	Request ID if the Trade Capture Report is in response to a Trade Capture Report Request	String
570	PreviouslyReported	N	Indicates if the trade capture report was previously reported to the counterparty. Not supported.	Boolean
748	TotNumTradeReports	N	Number of trade reports returned – if this report is part of a response to a Trade Capture Report Request	Int
828	TrdType	N	Type of Trade. Either 0 (Regular Trade) or 22 (Private Negotiated Deal).	Int
574	MatchType	N	The point in the matching process at which this trade was matched.	String
1300	MarketSegmentID	N	Identifies the market segment	String
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields	
Component block <YieldData>		N	Insert here the set of "YieldData" fields	
15	Currency	N	Primary currency of the specified currency pair. Used to qualify LastQty and GrossTradeAmount	Currency
31	LastPx	Y	Trade Price	Price
32	LastQty	Y	Trade Quantity	Qty
60	TransactTime	N	Time the transaction represented by this Trade Capture Report occurred	UTCTimeStamp
64	SettlDate	N	Specific date of trade settlement (Settlement Date) in YYYYMMDD format	LocalMktDate
75	TradeDate	N	Used when reporting other than current day trades	LocalMktDate
Start of Component block, expanded in line < TrdCapRptSideGrp >				

552	NoSides		Y	Number of sides	Int
→	54	Side	Y	Side of order	Char
→	37	OrderID	N	OrderID should be conditionally required when Trade Capture Report is used for back office processing	String
→	11	ClOrdID	N	Required for executions against electronically submitted orders which were assigned an ID by the institution or intermediary. In the case of quotes can be mapped to: - QuoteMsgID(1166) of a single Quote - QuoteID(117) of a Mass Quote	String
→	Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields	
→	1	Account	Y/N	May be required for electronically submitted orders which were assigned an account.	String
→	1093	LotType	N	Used to indicate sell side is a Block Divestment (value '3')	Char
→	58	Text	N	Free format text string	String
→	159	AccruedInterestAmt	N	Amount of Accrued Interest for convertible bonds and fixed income	Amt
End of Component block, expanded in line < TrdCapRptSideGrp >					
1390	TradePublishIndicator		N	Indicates if a trade in on or off market. 0 – Off market 1 – On market	Boolean
797	CopyMsgIndicator		N	Indicates Drop Copy	Boolean
StandardTrailer			Y		

5.5 Trade Capture Report Ack (AR)

The Trade Capture Report Ack message can be:

- Used to acknowledge trade capture report received
- Used to reject a trade capture report received

Table 24 – Trade Capture Report Ack

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AR	
571	TradeReportID	Y	Identifier for the Trade Capture Report	String
487	TradereportTransType	N	Identifier the Trade Report transaction type	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
856	TradeReportType	N	1 = Alleged 2 = Accept 3 = Decline	Int
150	ExecType	N	'F' - Trade	Char
939	TrdRptStatus	N	0 = Accepted 1 = Rejected	Int
17	ExecID	N	Exchanged assigned Execution ID (Trade Identifier)	String
751	TradeReportRejectReason	N	Reason for Rejection of Trade Report	int
572	TradeReportRefID	N	The TradeReportID that is being referenced for some action, such as acceptance or rejection	String
32	LastQty	N	Quantity for this Trade	Qty
31	LastPx	N	Price for this Trade	Price
48	SecurityID	Y	SecurityID for this Trade	String
58	Text	N	Text of reason if rejection or additional information if acception.	String
StandardTrailer		Y		

5.6 Trade Capture Report Request Ack (AQ)

The Trade Capture Report Request Ack message can be:

- Indicate that no trades were found that matched the selection criteria specified on the Trade Capture Report Request
- The Trade Capture Request was invalid for some business reason, such as request is not authorized, invalid or unknown instrument, party, trading session, etc.

NOTE: A Trade Capture Report Request Ack is not required if one or more Trade Capture Reports will be returned in-band immediately. Used to acknowledge trade capture reports received from a counterparty

Table 25 – Trade Capture Report Request Ack

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AQ	
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default.	Char
568	TradeRequestID	Y	Identifier for the trade request	String
569	TradeRequestType	Y	Type of Trade Capture Report.	Int
748	TotNumTradeReports	N	Number of trade reports returned	Int
749	TradeRequestResult	Y	Result of Trade Request.	Int
750	TradeRequestStatus	Y	Status of Trade Request.	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields	
58	Text	N	Free format text string	String
StandardTrailer		Y		

6 Market Data

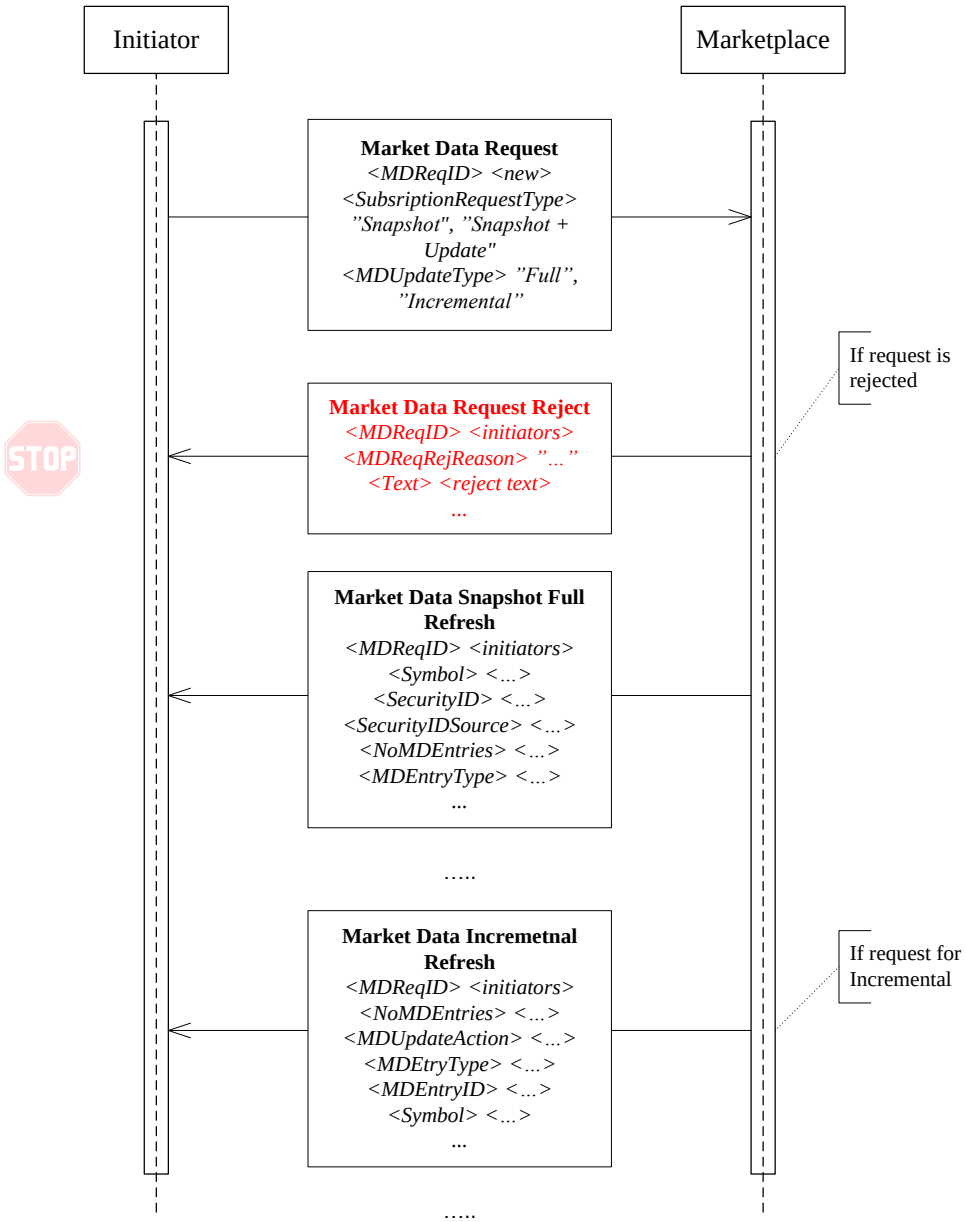
The market category consists of the following messages:

- Market Data Request
- Market Data Request Reject
- Market Data Snapshot/Full Refresh
- Trading Session Status Request
- Trading Session Status
- Security Status Request
- Security Status
- News

6.1 Workflows

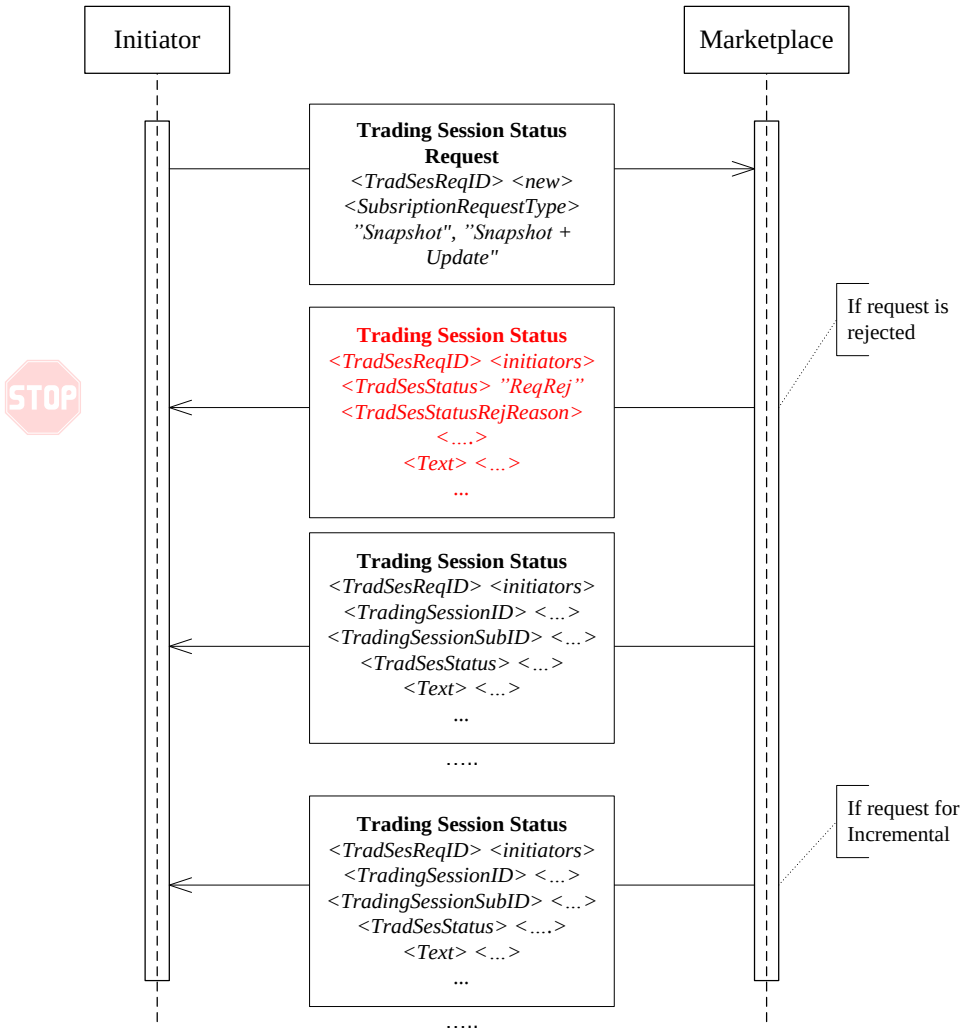
6.1.1 Subscribing to and Receiving Market Data

Figure 16 – Subscribing and Receiving Market Data Workflow



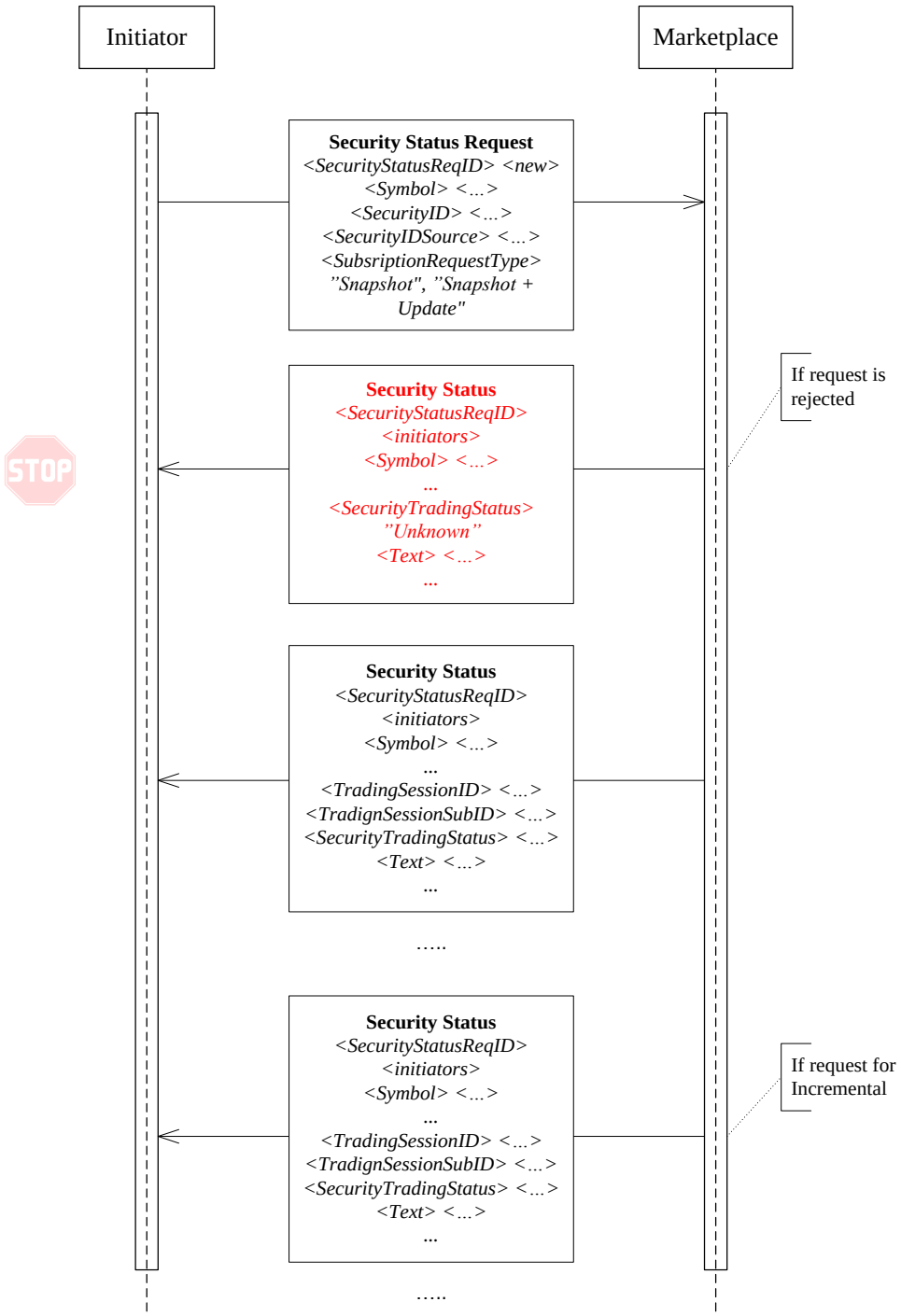
6.1.2 Subscribing to and Receiving Trading Session Status

Figure 17 – Subscribing and Receiving Trading Session Status Workflow



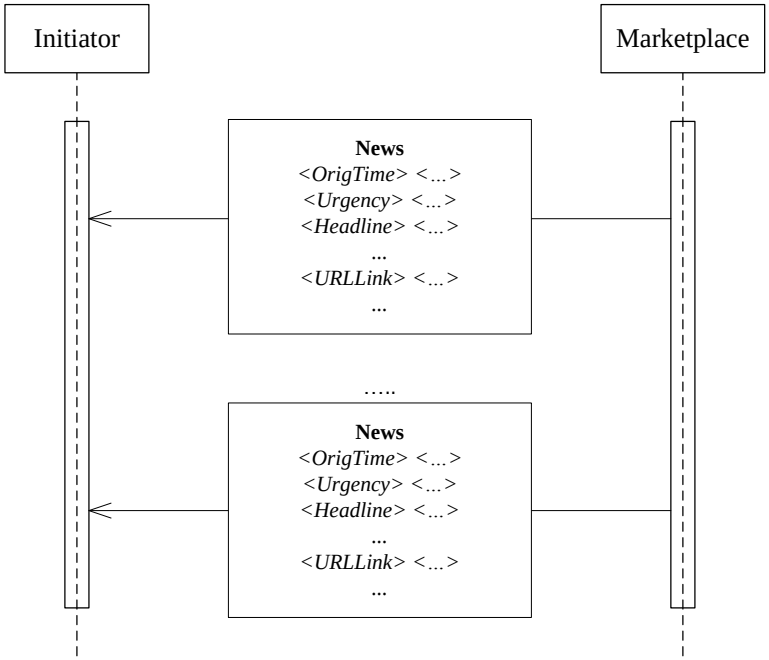
6.1.3 Subscribing to and Receiving Security Status

Figure 18 – Subscribing to and Receiving Security Status Workflow



6.1.4 Receiving News

Figure 19 – Receiving News Workflow



6.2 General Information for Market Data Requests and Responses

6.2.1 Market Data Requests Based on Category

Specific market data requests can be specified for the following major categories using the MDEntryType (269) tag:

- 0 - Order information – requests all order related information in market by order and market by price messages (e.g. bids, offers, etc.).
- 2 - Trade information – returns all market trade information and statistics.
- 3 - Index information – returns all index related information.
- 'a' - Security statistics (NSE specific) – returns security specific market statistics.

6.2.2 Market Data Snapshot Requests

Specific securities can be specified for snapshot requests with a configurable limit on the maximum number of securities. The default limit is configured to five (5) securities. Refer to the NSE FIX Installation Guide for details on this parameter. It should be also noted that the greater the number of securities specified the greater bandwidth and latency will be impacted.

6.2.3 Wildcard Security Specification

The '*' symbol is used to specify a market data request for all securities on one or more trading boards. The '*' symbol cannot be used within a security specification to request securities starting with, ending with or containing alphanumeric combinations such as 'AB*', '*AB' or 'A*B'.

6.2.4 Scenarios

The following scenarios will help clarify market data requests.

To get market data on all securities on all boards specify:

- NoRelatedSymbols (146) =1
- SecurityIDSource (22) = '99' and SecurityID (48)='*'
- SecuritySubType (762) is not specified.

To get market data on all securities on a single trading board specify:

- NoRelatedSymbols (146) =1
- SecurityIDSource (22) = '99' and SecurityID (48)='*'
- SecuritySubType (762) = 'EQTY'

To get market data on two selected securities on a single board:

- NoRelatedSymbols (146) =2
- SecurityIDSource (22) = '99' and SecurityID (48)='ABC001'
- SecurityIDSource (22) = '99' and SecurityID (48)='DEF888'
- SecuritySubType (762) = 'EQTY'

6.2.5 Response Enumerations Extensions

There are additional NSE specific enumerations which have been added for the MDEntryType (269) tag which apply only to market data request responses (not requests) and are used to identify the type of information being returned. Refer to Appendix C and the tables below for details on enumerated values for MDEntryType (269) for market data responses.

6.2.6 Order Information Request and Responses

The following table applies to Market Data requests for Order Information (MDEntryType = '0') and indicates which tags are used to return the requested information.

The FIX Gateway will send out 100 levels for both MarketByPrice and MarketByOrder.

Table 26 - Order Information Response Tags

Order Information (Request MDEntryType = '0')			
Returned MDEntryType (269)	Returned Tags	Tag Names	Description
Market By Order			
0,1	236	Yield	Bid/Offer Yield (Fixed Income securities only)
0,1	269	MDEntryType	Bid/Offer
0,1	270	MDEntryPx	Bid/Offer Price
0,1	271	MDEntrySize	Bid/Offer Quantity
0,1	272	MDEntryDate	Returned for Order Depth
0,1	273	MDEntryTime	Returned for Order Depth
0,1	290	MDEntryPositionNo	Display position of bid/offer
0,1	37	OrderID	Order ID
Market By Price			
0,1	236	Yield	Yield (Fixed Income securities only)
0,1	290	MDEntryPositionNo	Display position of bid/offer

0,1	269	MDEntryType	Bid/Offer
0,1	270	MDEntryPx	Bid/Offer Price
0,1	271	MDEntrySize	Total visible quantity of all orders at this price.
0,1	346	NumberOfOrders	Number of orders in the market

6.2.7 Trade Information Request and Responses

The following table applies to Market Data requests for Trade Information (MDEntryType = '2') and indicates which tags are used to return the requested information.

By default, only the last 10 trades are reported in a Market Data Snapshot/Full Refresh of trades for a security.

Table 27 - Trade Information Returned Tags

Trade Information (Request MDEntryType = '2')			
Returned MDEntryType (269)	Returned Tags	Tag Names	Description
2	31	LastPx	Value of this trade (price*quantity)
2	236	Yield	Yield (Fixed Income securities only)
2	270	MDEntryPx	Trade price
2	273 & 272	MDEntryDate, MDEntryTime	Both MDEntryDate(272) and MDEntryTime(273) are supplied.
2	1020	TradeVolume	Number of shares traded
2	279	MDUpdateAction	Trade Status - Matched, Cancelled, Amended. In IncrementalRefresh MDUpdateAction(279) indicates whether a trade is being matched/cancelled/amended. This tag is not provided on Snapshot/FullRefresh as 'cancelled' marketTrades are not disseminated.

6.2.8 Index Information Request and Responses

The following table applies to Market Data requests for Index Information (MDEntryType = '3') and indicates which tags are used to return the requested information.

Table 28 - Index Information Returned Tags

Index Information (Request MDEntryType = '3')			
Returned MDEntryType (269)	Returned Tags	Tag Names	Description
3	31	LastPx	Last index value
3	235, 236	YieldType (string = High), Yield	Trading session high yield*
3	235, 236	YieldType (string = Low), Yield	Trading session low yield*
3	235, 236	YieldType (string = Last), Yield	Last index yield*
3	235, 236	YieldType (string = OpenAvg), Yield	Opening Index Yield*

Index Information (Request MDEntryType = '3')			
3	235, 236	YieldType (string = Change), Yield	Change from reference yield*
4	270	MDEntryPx	Trading session opening Index price
7	270	MDEntryPx	Trading session high price
8	270	MDEntryPx	Trading session low price
t	270	MDEntryPx	Delta - Change from reference price
x	270	MDEntryPx	Reference Price
v	270	MDEntryPx	Total value of security traded today
w	1020	TradeVolume	Total number of shares traded today
y	332	HighPx	52 week high
y	333	LowPx	52 Week low

*Only returned if Yield values are applicable to fixed income securities/index.

6.2.9 Security Statistics Requests and Responses

The following table applies to Market Data requests for Security Statistics Information (MDEntryType = 'a') and indicates which tags are used to return the requested information.

Table 29 - Security Statistics Returned Tags

SECURITY STATISTICS (REQUEST MDENTRYTYPE = 'A')			
Returned MDEntryType (269)	Returned Tags	Tag Names	Description
4	270	MDEntryPx	Opening Price
4	286	OpenCloseSettlFlag	Official Opening Price Flag
5	270	MDEntryPx	Closing Price
5	286	OpenCloseSettlFlag	Official Closing Price Flag
6	270	MDEntryPx	Settlement Price
7	270	MDEntryPx	Trading session high price
8	270	MDEntryPx	Trading session low price
9	270	MDEntryPx	Weighted Average Price
a	31	LastPx	Last traded price
a	64	SettlDate	Specific date of trade settlement
a	274	TickDirection	Movement indicator
a	326	SecurityTradingStatus	Status of security
a	451	NetChgPrevDay	Change from previous day (for Market Data Incremental Refresh)

SECURITY STATISTICS (REQUEST MDENTRYTYPE = 'A')			
a	235, 236	YieldType, Yield	Trading session yields* Valid Return Values: High, Low, Last, WAvg (Weighted Average), OpenAvg (Open Average), Close
C	271	MDEntrySize	Open Interest
A	271	MDEntrySize	Size of the Imbalance (negative value indicates imbalance on the sell side, otherwise, imbalance is on the buy side).
n	271	MDEntrySize	Total number of trades today
o	271	MDEntrySize	Opening Quantity
s	270	MDEntryPx	Unadjusted Previous closing price.
t	270	MDEntryPx	Delta - Change from reference price
u	270	MDEntryPx	Previous closing price
v	270	MDEntryPx	Total value of security traded today
w	1020	TradeVolume	Total number of shares traded today
x	270	MDEntryPx	Reference Price
y	332	HighPx	52 week high price
y	333	LowPx	52 Week low price
z	270	MDEntryPx	Indicative Opening Price
z	271	MDEntrySize	Indicative Opening Quantity
z	336	TradingSessionID	SessionName
p	270	MDEntryPx	Indicative Closing Price
p	271	MDEntrySize	Indicative Closing Quantity
p	336	TradingSessionID	SessionName

*Only returned if Yield values are applicable to fixed income securities.

6.3 Market Data Request (V)

A successful Market Data Request returns one or more Market Data messages containing one or more Market Data Entries. Each Market Data Entry is a Bid, an Offer, a Trade associated with a security, the opening, closing, or settlement price of a security, the buyer or seller, the value of an index, the trading session high price, low price, or VWAP, or the trade volume or

open interest in a security. Market Data Entries usually have a price and a quantity associated with them. The market data request message format is as follows.

Table 30 – Market Data Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = V	
262	MDReqID	Y	Must be unique, or the ID of previous Market Data Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).	String
263	SubscriptionRequestType	Y	SubscriptionRequestType indicates to the other party what type of response is expected. A snapshot request only asks for current information. A subscribe request asks for updates as the status changes. Unsubscribe will cancel any future update messages from the counter party.	Char
264	MarketDepth	Y	Depth of market for Book Snapshot / Incremental updates. For market by price (MBP) this is limited to a depth of 100. For market by order (MBO) or non-aggregated book, this is limited to 100.	Int
265	MDUpdateType	N	Required if SubscriptionRequestType = Snapshot + Updates (1). Specifies the type of Market Data update.	Int
266	AggregatedBook	N	Specifies whether or not book entries should be aggregated. 'Y' = Market by Price (MBP), 'N' = Market by Order (MBO) – default.	Boolean
Start of Component block, expanded in line < MDReqGrp >				
267	NoMDEntryTypes	Y	Number of MDEntryType fields requested.	NumInGrp
→	269 MDEntryType	Y	Must be first field in repeating group. This is a list of all the category of Market Data Entries that the firm requesting the Market Data is interested in receiving.	Char
End of Component block, expanded in line < MDReqGrp >				
Start of Component block, expanded in line < InstrmtMDReqGrp >				
146	NoRelatedSym	Y	Number of securities requested. Must be greater than 0.	Int
→	Component block <Instrument>	Y	Insert here the set of "Instrument" (symbology) fields.	
End of Component block, expanded in line < InstrmtMDReqGrp >				
StandardTrailer		Y		

6.4 Market Data Request Reject (Y)

The Market Data Request Reject is used when the Exchange cannot honour the Market Data Request, due to business or technical reasons. The Exchange may choose to limit various

parameters, such as the size of requests, whether just the top of book or the entire book may be displayed, and whether Full or Incremental updates must be used.

The market data request reject message format is as follows.

Table 31 – Market Data Request Reject

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = Y	
262	MDReqID	Y	Must refer to the MDReqID of the request.	String
281	MDReqRejReason	N	Reason for the rejection of a Market Data request.	Char
58	Text	N	Free format text string	String
StandardTrailer		Y		

6.5 Market Data Snapshot/Full Refresh (W)

The Market Data messages are used as the response to a Market Data Request message. In all cases, one Market Data message refers only to one Market Data Request. Market Data messages sent as the result of a Market Data Request message will specify the appropriate MDReqID.

There are two types of Market Data Refresh messages, Snapshot/Full and Incremental.

The Market Data message format used for a Snapshot, or a Snapshot + Updates where MDUpdateType = Full Refresh (0) is as follows:

- After a Market Data Request, when a Bid or Offer is added, changed, or deleted, every update to a Market Data Entry results in a new Market Data Snapshot message that contains the entirety of the data requested for that instrument, not just the changed Market Data Entry. In other words, both sides of the market, or just one side in the case of a request of only bids or offers, for the depth requested, must be sent in one FIX Market Data Snapshot message.
- A Market Data Snapshot message may contain several trades, an index value, opening, closing, settlement, high, low, and/or VWAP price for one instrument, as well as the traded volume and open interest, but only for one instrument per message.
- Messages containing bids and/or offers cannot contain trades, index value, opening, closing, settlement, high, low, and/or VWAP prices, trade volume, or open interest.
- Messages containing Price Depth or Order Depth information for instruments traded in yield, the Yield component block will only contain Yield(236). YieldType(235) will not be sent in messages containing Price Depth or Order Depth information.

Table 32 – Market Data Snapshot/Full Refresh

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = W	
1021	MDBookType	N	Describes the type of book for which the feed is intended. Used when multiple feeds are provided over the same connection	Int
264	MarketDepth	N	Can be used to define the current depth of the book. For Market by Price (MBP) – aggregated book – this is limited to a depth of 100. For	Int

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				Market by Order (MBO) or non-aggregated book, this is limited to 100.	
75	TradeDate		N	Used to specify the trading date for which a set of market data applies	LocalMktDate
262	MDReqID		Y/N	Conditionally required if this message is in response to a Market Data Request.	Sting
Component block <Instrument>			Y	Insert here the set of "Instrument" (symbology) fields.	
451	NetChgPrevDay		N	Net change from previous day's closing price.	PriceOffset
Start of Component block, expanded in line < MDFullGrp >					
268	NoMDEntries		Y	Number of entries following.	NuminGroup
→	269	MDEntryType	Y	Must be first field in repeating group. This is a list of the type of Market Data Entries that the firm is receiving in this market data response message.	Int
→	278	MDEntryID	Y/N	Unique Market Data Entry identifier. Conditionally required when maintaining an order-depth book, that is, when AggregatedBook (266) is "N". This allows subsequent Incremental changes to be applied using MDEntryID.	String
→	Component block <YieldData>		N	Insert here the set of YieldData (yield-related) fields	
→	270	MDEntryPx	Y/N	Price of the Market Data Entry. Conditionally required depending on MDEntryType.	Price
→	271	MDEntrySize	Y/N	Quantity or volume of the market data entry. Conditionally required if MDEntryType = Trade(2). If MDEntryType = 2 (Trade) then TradeVolume (1020) will be populated.	Qty
→	272	MDEntryDate	N	Date of Market Data Entry.	UTCDateOnly
→	273	MDEntryTime	N	Time of Market Data Entry.	UTCTimeOnly
→	274	TickDirection	N	Direction of the "tick".	Char
→	286	OpenCloseSettlFlag	N	Official Open/Close Price Flag. Values: 0 – Indicates Official Open/Close	MultipleCharValue

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
→	326	SecurityTradingStatus	N	Identifies the trading status applicable to the transaction.	Int
→	290	MDEntryPositionNo	N	Display position of a bid or offer, numbered from most competitive to least competitive, per market side beginning with 1.	Int
→	346	NumberOfOrders	N	In an Aggregated Book, used to show how many individual orders make up an MDEntry	Int
→	332	HighPx	N	Highest price paid for the security in the trading session. If MDEntryType = 'y' then this value is the 52 week high price.	Price
→	333	LowPx	N	Lowest price paid for the security in the trading session. If MDEntryType = 'y' then this value is the 52 week low price.	Price
→	31	LastPx	N	Price of this fill. (Note: The LastPx field first appears in market data messages in FIX V5.0 SP2 but is included in this specification for completeness).	Price
→	1020	TradeVolume	N	Used to report trade volume in association with trade, bid or ask rather than a separate entity. This is a V5.0 tag value.	Int
→	64	SettlDate	N	Specific date of trade settlement (Settlement Date) in YYYYMMDD format.	LocalMktDate
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < MDFullGrp >					
StandardTrailer			Y		

6.6 Market Data Incremental Refresh (X)

The second Market Data message format is used for incremental updates. With the incremental message the Exchange has the responsibility to provide all Market Data entries needed by the client user in order to build an order book copy, populate a Trade Ticker, etc.

The Market Data Incremental Refresh message may contain any combination of new, changed, or deleted Market Data Entries, for one or more instruments, with any combination of trades, imbalances, quotes, index values, open, close, settlement, high, low, and VWAP prices, trade volume and open interest so long as the maximum FIX message size is not exceeded.

Market Data Entries may have an MDEntryID unique among all currently active Market Data Entries so they can be referenced for the purposes of deleting and changing them later. When changing a Market Data Entry, it may keep the same MDEntryID, in which case only MDEntryID would be populated, or the MDEntryID may change, in which case MDEntryID will

contain the new ID and MDEntryRefID will contain the ID of the Market Data Entry being changed. An MDEntryID can be reused within a day only if it has first been deleted.

6.6.1 Maintaining the Order Book

The following instructions (MDUpdateAction) are used to maintain the order book:

- New – Used to insert a market data entry.
- Delete – Used to remove a market data entry or to remove all entries from the order book

  When an order book gets empty during a trading day, a Market Data Incremental Refresh message will be sent to indicate the removal of all entries from the order book as follows:

  NoMDEntries(268)=1 MDUpdateAction=2(Delete) and MDEntryType=J(EmptyBook)
- Change – Used to modify a market data entry.

Table 33 – Market Data Incremental Refresh

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = X	
262	MDReqID		Y/N	Conditionally required if this message is in response to a Market Data Request.	String
1021	MDBookType		N	Describes the type of book for which the feed is intended. Used when multiple feeds are provided over the same connection	Int
Start of Component block, expanded in line < MDIncGrp >					
268	NoMDEntries		Y	Number of entries following.	NumInGrp
→	279	MDUpdateAction	Y	Type of update action. Must be first field in this repeating group.	Char
→	264	MarketDepth	N	Can be used to define the current depth of the book. For Market by Price (MBP) - aggregated book – this is limited to a depth of 100. For Market by Order (MBO) or non-aggregated book, this is limited to 100.	Int
→	269	MDEntryType	N	Must be first field in repeating group. This indicates the type of Market Data Entries that the firm is receiving in this market data response message.	Int
→	278	MDEntryID	Y/N	If specified, must be unique among currently active entries if MDUpdateAction = New (0), must be the same as a previous MDEntryID if MDUpdateAction = Delete (2), and must be the same as a previous MDEntryID if MDUpdateAction = Change (1) and MDEntryRefID is not specified, or must be unique among currently	String

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				active entries if MDUpdateAction = Change(1) and MDEntryRefID is specified. Conditionally required when maintaining an order-depth book, that is, when AggregatedBook (266) is "N". This allows subsequent Incremental changes to be applied using MDEntryID.	
→	Component block <Instrument>		N	Insert here the set of "Instrument" (symbology) fields. Either Symbol (the instrument component block) or MDEntryRefID must be specified if MDUpdateAction = New(0) for the first Market Data Entry in a message. For subsequent Market Data Entries where MDUpdateAction = New(0), the default is the instrument used in the previous Market Data Entry.	
→	270	MDEntryPx	Y/N	Price of the Market Data Entry. Conditionally required when MDUpdateAction = New(0) and MDEntryType is not Trade Volume (B).	Price
→	Component block <YieldData>		N	Contains Yield information.	
→	271	MDEntrySize	Y/N	Quantity or volume represented by the Market Data Entry. Conditionally required when MDUpdateAction = New(0) and MDEntryType Trade(2). If MDEntryType = 2 (Trade) then TradeVolume (1020) will be populated.	Qty
→	272	MDEntryDate	N	Date of Market Data Entry.	UTCDateOnly
→	273	MDEntryTime	N	Time of Market Data Entry.	UTCTimeOnly
→	274	TickDirection	N	Direction of the "tick".	Char
→	286	OpenCloseSettlFlag	N	Official Open/Close Price Flag. Values: 0 – Indicates Official Open/Close	MultipleCharValue
→	333	LowPx	N	Lowest price paid for the security in the trading session. If MDEntryType = 'y' then this value is the 52 week low price.	Price
→	31	LastPx	N	Price of this fill. (Note: The LastPx field first appears in market data messages in FIX V5.0 SP2 but is included in this specification for completeness).	Price

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
→	451	NetChgPrevDay	N	Net change from previous day's closing price.	PriceOffset
→	1020	TradeVolume	N	Used to report trade volume in association with trade. This is a V5.0 tag value.	Int
→	326	SecurityTradingStatus	N	Identifies the trading status applicable to the transaction.	Int
→	346	NumberOfOrders	N	In an Aggregated Book, used to show how many individual orders make up an MDEntry	Int
→	290	MDEntryPositionNo	N	Display position of a bid or offer, numbered from most competitive to least competitive, per market side, beginning with 1	Int
→	64	SettlDate	N	Specific date of trade settlement (Settlement Date) in YYYYMMDD format.	LocalMktDate
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < MDIncGrp >					
StandardTrailer			Y		

6.7 Trading Session Status Request (g)

The Trading Session Status Request is used to request information on the status of a market. With the move to multiple sessions occurring for a given trading party (morning and evening sessions for instance) there is a need to be able to provide information on what product is trading on what market.

The Trading Session Status Request message can be used to inquire the trading status of a trading party. The Trading Session Status message can be used to subscribe to updates to the status of a trading session by setting the RequestType field to 1.

The following rule applies for Trading Session Status Requests with snapshot only (SubscriptionRequestType = 0):

- If the MarketSegmentID tag (1300) is set to an X-stream Board ID then only the trading session information for that Board will be returned otherwise trading session information for all Boards will be returned.

Table 34 – Trading Session Status Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = g (lowercase)	
335	TradSesReqID	Y	Must be unique, or the ID of previous Trading Session Status Request to disable if SubscriptionRequestType = Disable previous Snapshot+Updates Request (2).	String
1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag	Exchange

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			value.	
1300	MarketSegmentID	N	Market Segment for which Trading Session applies. This is a V5.0 tag value.	String
263	SubscriptionRequestType	Y	Subscription type request.	Char
StandardTrailer		Y		

6.8 Trading Session Status (h)

The Trading Session Status provides information on the status of a market. For markets multiple trading sessions on multiple-markets occurring (morning and evening sessions for instance), this message is able to provide information on what products are trading on what market during what trading session.

Table 35 – Trading Session Status

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = h (lowercase)	
335	TradSesReqID	N	Provided for a response to a specific Trading Session Status Request message (snapshot).	String
336	TradingSessionID	Y	Identifier for Trading Session. A trading session spans an extended period of time that can also be expressed informally in terms of the trading day. Usage is determined by market or counterparties. In the context of this message the trading session ID will map to the current trading session state (e.g. pre-open, open, closed, etc.)	String
1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment for which Trading Session applies. This is a V5.0 tag value.	String
340	TradSesStatus	Y	State of the trading session	Int
567	TradSesStatusRejReason	N	Use with TradSesStatus = "Request Rejected"	Int
341	TradSesStartTime	N	Starting time of the trading session	UTCTimeStamp
58	Text	N	Free format text string	String
StandardTrailer		Y		

6.9 Security Status Request (e)

The Security Status Request message provides for the ability to request the status of a security. One or more Security Status messages are returned as a result of a Security Status Request message.

The Security Status Request message contains a *SubscriptionRequestType* field. This tells the counter party what type of request is being made:

- 0 – indicates that the requestor only wants a snapshot or the current status.
- 1 – indicates that the requestor wants a snapshot (the current status) plus updates as the status changes. This is similar to subscribing for information and can be implemented in applications as a subscription mechanism.
- 2 – indicates that the requestor wishes to cancel any pending snapshots or updates – in essence making this an unsubscribe operation.

The following rules apply for Security Status Requests with snapshot only (SubscriptionRequestType = 0):

- If a component block instrument is sent with the tag 1300 (MarketSegmentID) set to an X-stream Board ID then the security status information is sent only for the specified symbol from the specified Board.
- If the combination of Board ID and security ID is not found a reject is returned.

Table 36 – Security Status Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = e (lowercase)	
263	SubscriptionRequestType	Y	SubcriptionRequestType indicates to the other party what type of response is expected. A snapshot request only asks for current information. A subscribe request asks for updates as the status changes. Unsubscribe will cancel any future update messages from the counter party. Subscribe or unsubscribe for security status for security specified in request.	Char
324	SecurityStatusReqID	Y	Must be unique, or the ID of previous Security Status Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).	String
component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
1301	MarketID	N	Identifies the Market – MIC for the market conforming to ISO10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security trades. It is mapped to X-stream Board Id. This is a V5.0 tag value.	String
StandardTrailer		Y		

6.10 Security Status (f)

The Security Status message provides for the ability to report changes in status to a security. The Security Status message is used by the Exchange to report changes in the state of a security.

Table 37 – Security Status

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = f (lowercase)	

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
324	SecurityStatusReqID	N		String
component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
15	Currency	N	Identifies the currency used for price. Absence of this field is interpreted as the default for the security as defined in the reference data.	Currency
31	LastPx	N	Represents the last price for that security either on a Consolidated or an individual participant basis at the time it is disseminated. For Fixed Income securities this will be the Last Yield expressed as a percentage value.	Price
60	TransactTime	N	Time of dissemination	UTCTimeStamp
292	CorporateAction	N	Identifies the type of Corporate Action (if applicable). Also referred to as 'Basis of Quotation'.	MultipleCharValue
326	SecurityTradingStatus	Y	Identifies the trading status applicable to the security.	Int
333	LowPx	N	Low price. For Fixed Income securities this will be the Low Yield expressed as a percentage value.	Price
1174	SecurityTradingEvent	N	Identifies an event related to a SecurityTradingStatus(326). When a Circuit Breaker triggers, causing a security to be suspended, tag 1174=3 will be disseminated.	int
332	HighPx	N	High price. For Fixed Income securities this will be the High Yield expressed as a percentage value.	Price
1301	MarketID	N	Identifies the Market where the security is trading (e.g. XASX, XSGX, etc.). This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security trades. It is mapped to X-stream Board Id. This is a V5.0 tag value.	String
58	Text	N	Free format text string	String
Standard Trailer		Y		

6.11 News (B)

The news message is a general free format message between the participant and Exchange. The message contains flags to identify the news item's urgency and to allow sorting by Subject Company (symbol). The News message can be originated at either the broker or institution side.

Table 38 – News

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = B	
42	OrigTime	N	Time if message origination. Always expressed in UTC time.	UTCTimeStamp
61	Urgency	N	Urgency Flag.	Char
148	Headline	Y	Specifies the headline text	String
358	EncodedHeadlineLen	N	Byte length of encoded (non-ASCII characters) EncodedHeadline (359) field.	Length
359	EncodedHeadline	N	Encoded (non-ASCII characters) representation of the Headline (148) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation will also be specified in the Headline field.	Data
Start of Component block, expanded in line < InstrmtGrp >				
146	NoRelatedSym	N	Specifies the number of repeating symbols (instruments) specified	NumInGroup
→	component block <Instrument>	N	Insert here the set of "Instrument" (symbology)	
End of Component block, expanded in line < InstrmtGrp >				
component block <LinesOfTextGroupt>		Y	Insert here the set of "LinesOfTextGroup" fields.	
149	URLLink	N	A URL (Uniform Resource Locator) link to additional information (i.e. http://www.XYZ.com/research.html)	String
StandardTrailer		Y		

7 Reference Data

The reference data category consists of the following messages:

- Market Definition Request
- Market Definition
- Trading Session List Request
- Trading Session List
- Trading Session List Update Report
- Security List Request
- Security List
- Security List Update
- Security Definition Request
- Security Definition
- Security Definition Update

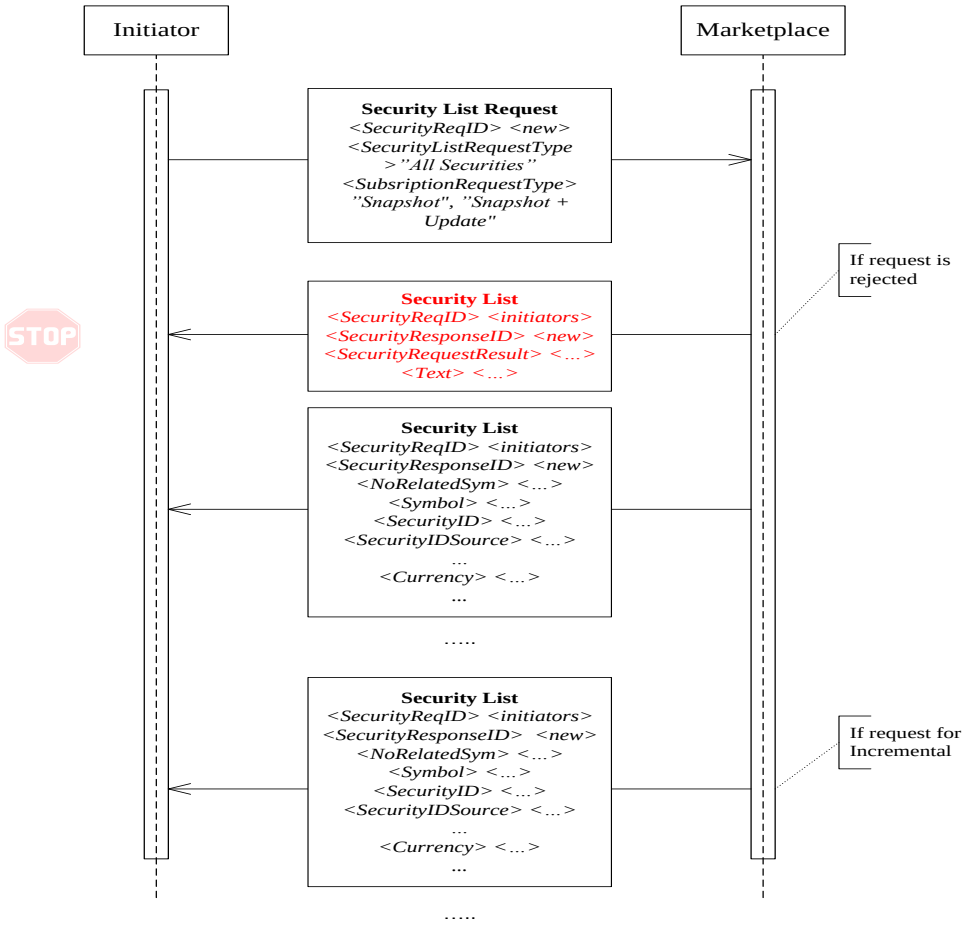
7.1 Workflows

The following is the workflow for

- requesting and receiving a list of securities and
- 'start of day' download of relevant reference data.

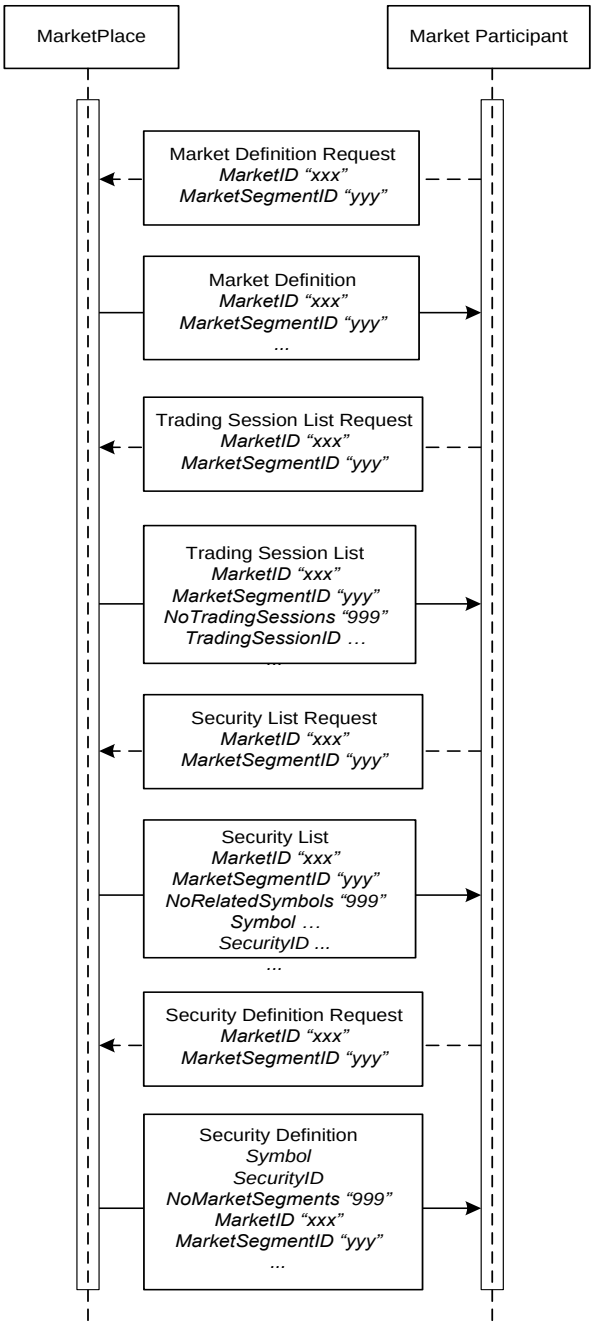
7.1.1 Requesting and Receiving a list of securities workflow

Figure 20 – Requesting and Receiving a List of Securities



7.1.2 'Start of Day' download workflow

Figure 21 – 'Start of Day' Download



7.2 Market Definition Request (BT)

The Market Definition Request message is used to request market structure information from the Exchange. Specified fields will act as "filters" for the request. For example, if MarketID is specified then only the market structure information for that specified market will be sent back if it is available. If the MarketID is not specified then the request is for all available market structure information.

The Market Definition Request can also indicate to the Exchange whether the request is for a snapshot of requested information, subscribe to market structure information, or to unsubscribe to an earlier subscription request. This is done via the SubscriptionRequestType (263) field.

Table 39 – Market Definition Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BT	
1393	MarketReqID	Y	Must be unique, or the ID of previous Market Segment Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request(2). This is a V5.0 tag value.	String
263	SubscriptionRequestType	Y	Subscribe or unsubscribe for security status to security specified in request. Subscription type request. Only snapshot (0) is supported.	Char
1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383 (e.g. XASX, XSGX, etc.). This is a V5.0 tag value.	String
1300	MarketSegmentID	N	Market Segment (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
StandardTrailer		Y		

7.3 Market Definition (BU)

The Market Definition message is used to respond to Market Definition Request. In a subscription, it will be used to provide the initial snapshot of the information requested.

This message is associated with a list of trading sessions applicable for the market segment – the list is published using the Trading Session List message. Refer to the workflow diagram in section 7.1.2.

Table 40 – Market Definition

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BU	
1301	MarketID	Y	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	String
1300	MarketSegmentID	N	Market Segment within the Market (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
1393	MarketReqID	N	Unique ID of the Market Definition Request Message. This is a V5.0 tag value.	String
1394	MarketReportID	Y	Unique identifier for each Market Definition message. This is a V5.0 tag value.	String
1396	MarketSegmentDesc	N	Describes the Market Segment. This is a V5.0 tag value. Not supported.	String
58	Text	N	Free format text string	String
Standard Trailer		Y		

7.4 Trading Session List Request (BI)

The Trading Session List Request is used to request a list of trading sessions available in a market place and the state of those trading sessions.

A successful request will result in a response from the counterparty of a Trading Session List (MsgType=BJ) message that contains a list of zero or more trading sessions.

The TradSesReqID must be used to provide a unique identifier for the request. This value is returned by the Exchange in the Trading Session List messages sent in response to the request.

The Trading Session List Request follows the standard request model in providing the SubscriptionRequestType (tag 263) field which can be used to obtain a snapshot of trading session information, subscribe for a snapshot with subsequent updates, or to unsubscribe from a previous subscription request.

Table 41 – Trading Session List Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BI	
263	SubscriptionRequestType	Y	Subscription type request.	Char
335	TradSesReqID	Y	Must be unique, or the ID of previous Trading Session Status Request to disable if SubscriptionRequestType = Disable previous Snapshot + Update Request (2).	String
1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment for which Trading Session applies (e.g. equities, debt, etc.). It is mapped to X-stream Board Id. This is a V5.0 tag value.	String
StandardTrailer		Y		

7.5 Trading Session List (BJ)

The Trading Session List message is sent as a response to a Trading Session List Request. The Trading Session List should contain the characteristics of the trading session(s) and the current state of the trading session(s).

The message could be relayed every trading day, or at least when trading sessions are changed. Depending on characteristics of the market, the various Time fields may apply.

The Trading Session List returns the TradSesReqID (tag 335) value from the Trading Session List Request originally sent by a counterparty.

Table 42 – Trading Session List

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BJ	
335	TradSesReqID	N	Provided for a response to a specific Trading Session List Request message (snapshot).	String
Start of Component block, expanded in line < TrdSessLstGrp >				

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
386	NoTradingSessions		Y	Number of TradingSessionIDs (336) in repeating group.	NumInGrp
→	336	TradingSessionID	Y	Identifier for Trading Session	String
→	340	TradSesStatus	Y	State of trading session.	Int
→	341	TradSesStartTime	N	Starting time of trading session	UTCTimeStamp
→	342	TradSesOpenTime	N	Time of the opening of the trading session	UTCTimeStamp
→	567	TradSesStatusRejReason	N	Used with TradSesStatus = "Request Rejected".	Int
→	1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
→	1300	MarketSegmentID	N	Market Segment for which Trading Session applies (e.g. equities, debt, futures, options, etc.). This is a V5.0 tag value.	String
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < TrdSessLstGrp >					
StandardTrailer			Y		

7.6 Trading Session List Update Report (BS)

The Trading Session List Update Report is used by marketplaces to provide intra-day updates of trading sessions when there are changes to one or more trading sessions.

Table 43 – Trading Session List update report

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = BS	
335	TradSesReqID		N	Provided for a response to a specific Trading Session List Request message (snapshot).	String
1327	TradSesUpdateAction		N	Specifies the action taken for the specified trading sessions. This is a V5.0 tag value.	Char
Start of Component block, expanded in line < TrdSessLstGrp >					
386	NoTradingSessions		Y	Number of TradingSessionIDs (336) in repeating group.	NumInGrp
→	336	TradingSessionID	Y	Identifier for Trading Session. In the context of this message, the trading session ID will map to the current trading session state (e.g. pre-open, open, closed, etc.)	String

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
→	340	TradSesStatus	Y	State of trading session.	Int
→	341	TradSesStartTime	N	Starting time of trading session	UTCTimeStamp
→	342	TradSesOpenTime	N	Time of the opening of the trading session	UTCTimeStamp
→	567	TradSesStatusRejReason	N	Used with TradSesStatus = "Request Rejected"	Int
→	1301	MarketID	N	This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
→	1300	MarketSegmentID	N	Market Segment for which Trading Session applies (e.g. equities, debt, futures, options, etc.). This is a V5.0 tag value.	String
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < TrdSessLstGrp >					
StandardTrailer			Y		

7.7 Security List Request (x)

The Security List Request message is used to return a list of securities from the Exchange that match criteria provided on the request. The SecurityListRequestType[559] tag specifies the criteria of the request.

Subscription request for security status can be optionally specified by including the SubscriptionRequestType[263] field on the message.

Table 44 – Security List Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = x (lowercase X)	
263	SubscriptionRequestType	N	Subscribe or unsubscribe for security status to security specified in request. Subscribe or unsubscribe for security status to security specified in request.	Char
320	SecurityReqID	Y	Unique ID for Security List Request	String
559	SecurityListRequestType	Y	Type of Security List Request being made	Int
1301	MarketID	N	Identifies the market which lists and trades the instrument. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security is traded (e.g. equities, debt, etc.). It is mapped to X-stream Board Id. This is a V5.0 tag value. Must be specified if <Instrument> block is entered.	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
Component block <Instrument>		Y/N	Insert here the set of "Instrument" (symbology) fields. Required if only a single security is requested.	
Standard Trailer		Y		

7.8 Security List (y)

The Security List message is used to return a list of securities that matches the criteria specified in a Security List Request (x).

The instrument component block in Security List will contain the following additional fields to provided non-English symbol and descriptions:

- EncodedSecurityDescLen(350)
- EncodedSecurityDesc(351)
- EncodedSymbolLen(1359)
- EncodedSymbol(1360)

If the tick rules do not change during the trading day, the NoTickRule block will not be included in the message.

Table 45 – Security List

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = y (lowercase Y)	
320	SecurityReqID	Y	Unique Id for the Security Definition Request.	String
322	SecurityResponseID	Y	Identifier for the Security List message	String
393	TotNoRelatedSym	N	Used to indicate the total number of securities being returned for this request. Used in the event that message fragmentation is required.	Int
560	SecurityRequestResult	Y	Result of the Security Request identified by the SecurityReqID.	Int
1301	MarketID	N	Identifies the market which lists and trades the instrument. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security is traded (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
893	LastFragment	N	Indicates whether this is the last fragment in a sequence of message fragments. When set this indicates the last of the message group.	Boolean
Start of Component block, expanded in line < SecListGrp >				
146	NoRelatedSym	Y/N	Specifies the number of repeating symbols (instruments) specified. Required if SecurityRequestResult (560) = 0 (Valid	NumInGrp

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				request)	
→	Component block <Instrument>		Y/N	Insert here the set of "Instrument" (symbology) fields. Required if SecurityRequestResult (560) = 0 (Valid request)	
→	15	Currency	N	Identifies the currency used for price. Absence of this field is interpreted as the default for the security as defined in the reference data.	Currency
→	1306	PriceLimitType	N	Describes the how the price limits are expressed 0 = Price	Int
→	1148	LowLimitPrice	N	Allowable low limit price for the trading day. A key parameter in validating order price. Used as the lower band for validating order prices. Orders submitted with prices below the lower limit will be rejected.	Price
→	1149	HighLimitPrice	N	Allowable high limit price for the trading day. A key parameter in validating order price. Used as the upper band for validating order prices. Orders submitted with prices above the upper limit will be rejected.	Price
→	1150	TradingReferencePrice	N	Reference price of the security	Price
→	1205	NoTickRules	N	Number of TickRules	NumInGroup
→	→	1206 StartTickPriceRange	N	Starting price range for specified tick increment	Price
→	→	1207 EndTickPriceRange	N	Ending price range for the specified tick increment	Price
→	→	1208 TickIncrement	N	Tick increment for stated price range. Specifies the valid price increments at which a security can be quoted and traded	Price
→	562	MinTradeVol	N	The minimum trading volume for a security	Qty
→	1140	MaxTradeVol	N	The maximum order quantity that can be submitted for a security.	Qty
→	Component block <YieldData>		N	Returns Yield information for Fixed Income securities.	
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < SecListGrp >					
StandardTrailer			Y		

7.9 Security List Update Report (BK)

The Security List Update Report is used for reporting updates to reference database. Updates could be due to Corporate Actions or other business events. Update may include additions, modifications and deletions.

Table 46 – Security List Update Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BK	
320	SecurityReqID	Y	Unique Id for the Security Definition Request.	String
322	SecurityResponseID	Y	Identifier for the Security Definition message.	String
393	TotNoRelatedSym	N	Used to indicate the total number of securities being returned for this request. Used in the event that message fragmentation is required.	Int
560	SecurityRequestResult	N	Result of the Security Request identified by the SecurityReqID. The results returned to a Security Request message.	Int
1301	MarketID	N	Identifies the market which lists and trades the instrument. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security is traded (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
893	LastFragment	N	Indicates whether this is the last fragment in a sequence of message fragments. When set this indicates the last of the message group.	Boolean
292	CorporateAction	N	Identifies the type of Corporate Action that triggered the update (if applicable). Also referred to as 'Basis of Quotation'.	MultipleCharValue
Start of Component block, expanded in line < SecLstUpdRelSymGrp >				
146	NoRelatedSym	Y	Specifies the number of repeating symbols (instruments) specified.	NumInGrp
→	1324 ListUpdateAction	N	If provided then the Instrument occurrence has explicitly changed. This is a V5.0 tag value.	Char
→	Component block <Instrument>	Y	Insert here the set of "Instrument" (symbology) fields.	
→	15 Currency	N	Identifies the currency used for price. Absence of this field is interpreted as the default for the	Currency

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				security as defined in the reference data.	
→	1306	PriceLimitType	N	Describes the how the price limits are expressed. 0 = Price	Int
→	1148	LowLimitPrice	N	Allowable low limit price for the trading day. A key parameter in validating order price. Used as the lower band for validating order prices. Orders submitted with prices below the lower limit will be rejected.	Price
→	1149	HighLimitPrice	N	Allowable high limit price for the trading day. A key parameter in validating order price. Used as the upper band for validating order prices. Orders submitted with prices above the upper limit will be rejected.	Price
→	1150	TradingReferencePrice	N	Reference price of the security	Price
→	1205	NoTickRules	N	TickRules	
→	→	1206 StartTickPriceRange	N	Starting price range for specified tick increment	Price
→	→	1207 EndTickPriceRange	N	Ending price range for the specified tick increment	Price
→	→	1208 TickIncrement	N	Tick increment for stated price range. Specifies the valid price increments at which a security can be quoted and traded	Price
→	562	MinTradeVol	N	The minimum trading volume for a security	Qty
→	1140	MaxTradeVol	N	The maximum order quantity that can be submitted for a security.	Qty
→	Component block <YieldData>		N	Insert here the set of "YieldData" fields.	
→	58	Text	N	Free format text string	String
End of Component block, expanded in line < SecLstUpdRelSymGrp >					
StandardTrailer			Y		

7.10 Security Definition Request (c)

The Security Definition Request message is used for the following:

- Request the definition of a specific security.
- Request the definitions for a set of individual securities for a single market segment.
- Request the definitions of all securities, independent of market segment.

Subscription for security status can be optionally specified by including the SubscriptionRequestType[263] field on the message.

Table 47 – Security Definition Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = c (lowercase)	
320	SecurityReqID	Y	Unique ID of a Security Definition Request	String
321	SecurityRequestType	Y	Type of Security Definition Request	Int
263	SubscriptionRequestType	N	Subscribe or unsubscribe for security status to security specified in request.	Char
1301	MarketID	Y/N	Identifies the market for which the security definition request is being made. Required if SecurityRequestType (321) = 9 (MarketID). This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
1300	MarketSegmentID	N	Market Segment where the security is traded (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
Component block <Instrument>		Y/N	Insert here the set of "Instrument" (symbology) fields. Required if SecurityRequestType (321) = 0 (Security identity and specifications), 1 (Security identity for specifications provided), or 4 (Symbol).	
Standard Trailer		Y		

7.11 Security Definition (d)

The Security Definition message is used for the following:

- Respond to a request for a within a specified market segment.
- Convey a comprehensive security definition for all market segments that the security participates in.

The instrument component block in Security Definition will contain the following additional fields to provided non-English symbol and descriptions:

- EncodedSecurityDescLen(350)
- EncodedSecurityDesc(351)
- EncodedSymbolLen(1359)
- EncodedSymbol(1360)

Table 48 – Security Definition

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = d (lowercase)	
320	SecurityReqID		Y	Unique ID for the Security Definition	String
322	SecurityResponseID		Y	Identifier for the Security Definition message	String
323	SecurityResponseType		Y	Type of Security Definition response.	Int
292	CorporateAction		N	Identifies the type of Corporate Action that may be associated with the security. Also referred to as 'Basis of Quotation'.	MultipleCharValue
Component block <Instrument>			Y	Insert here the set of "Instrument" (symbology) fields.	
15	Currency		N	Identifies the currency used for price. Absence of this field is interpreted as the default for the security as defined in the reference data master file.	Currency
Component block <YieldData>			N	Insert here the set of "YieldData" fields. Contains Yield information for Fixed Income securities.	
Start of Component block, expanded in line < UndInstrmtGrp >					
711	NoUnderlyings		N	Number of Underlying Instruments.	NumInGrp
→	309		N	Underlying security's SecurityID.	String
→	305		N	Underlying security's SecurityIDSource.	String
→	763		N	Underlying security's SecuritySubType.	String
End of Component block, expanded in line < UndInstrmtGrp >					
Start of Component block, expanded in line < MarketSegmentGrp >					
1310	NoMarketSegments		N	Number of Market Segments on which a security may trade. This is a V5.0 tag value.	NumInGrp
→	1301	MarketID	N	Identifies the market which lists and trades the instrument. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
→	1300	MarketSegmentID	N	Market Segment where the security is traded (e.g. equities, debt, etc.). This is a V5.0 tag value.	String
End of Component block, expanded in line < MarketSegmentGrp >					
58	Text		N	Free format text string	String
StandardTrailer			Y		

7.12 Security Definition Update Report (BP)

This message is used for reporting updates to the reference database. Updates could be the result of corporate actions or other business events. Updates may include additions, modifications or deletions. Only a single security is reported in each update message.

Table 49 – Security Definition Update Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BP	
320	SecurityReqID	Y	Unique ID for the Security Definition	String
322	SecurityResponseID	Y	Identifier for the Security Definition message.	String
980	SecurityUpdateAction	Y	Type of action.	Char
292	CorporateAction	N	Identifies the type of Corporate Action that triggered the update (if applicable). Also referred to as 'Basis of Quotation'.	MultipleCharValue
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
15	Currency	N	Identifies the currency used for price. Absence of this field is interpreted as the default for the security as defined in the reference data.	Currency
Component block <YieldData>		N	Insert here the set of "YieldData" fields.	
Start of Component block, expanded in line < UndInstrmtGrp >				
711	NoUnderlyings	N	Number of Underlying Instruments	NumInGrp
→ 309		N	Underlying security's SecurityID.	String
→ 305		N	Underlying security's SecurityIDSource.	String
→ 763		N	Underlying security's SecuritySubType.	String
End of Component block, expanded in line < UndInstrmtGrp >				
Start of Component block, expanded in line < MarketSegmentGrp >				
1310	NoMarketSegments	N	Number of Market Segments on which a security may trade. This is a V5.0 tag value.	NumInGrp
→ 1301	MarketID	N	Identifies the market which lists and trades the instrument. This is the Market Identification Code (MIC) conforming to ISO-10383. This is a V5.0 tag value.	Exchange
→ 1300	MarketSegmentID	N	Market Segment where the security	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			is traded (e.g. equities, debt, etc.). This is a V5.0 tag value.	
End of Component block, expanded in line < MarketSegmentGrp >				
58	Text	N	Free format text string	String
StandardTrailer		Y		

Appendix A - Standard Header and Trailer

A.1 Standard Header

The standard message header format is as follows.

Table 50 – Standard Message Header

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
8	BeginString	Y	FIXT.1.1 (always unencrypted, must be first field in message)	String
9	BodyLength	Y	(Always unencrypted, must be second field in message)	Length
35	MsgType	Y	(Always unencrypted, must be third field in message)	String
1128	ApplVerID	N	Specifies the service pack release being applied at the message level. The only valid value is '8' = FIX50SP1	String
49	SenderCompID	Y	(Always unencrypted). Identifies the firm sending the message.	String
56	TargetCompID	Y	(Always unencrypted). Identifies the firm receiving the message.	String
115	OnBehalfOfCompID	N	Trading partner company ID used when sending messages via a third party (Can be embedded within encrypted data section). Not supported.	String
116	OnBehalfOfSubID	N	Trading partner SubID used when delivering messages via a third party (Can be embedded within encrypted data section). Not supported.	String
144	OnBehalfOfLocationID	N	Trading partner LocationID (i.e. geographic location and/or desk) used when delivering messages via a third party. (Can be embedded within encrypted data section). Not supported.	String
128	DeliverToCompID	N	Trading partner company ID used when sending messages via a third party (Can be embedded within encrypted data section). Not supported	String
34	MsgSeqNum	Y	(Can be embedded within encrypted data section.)	SeqNum
50	SenderSubID	N	Assigned value used to identify specific message originator (e.g. desk, trader, etc.)	String
142	SenderLocationID	N	Sender's LocationID (i.e. geographic location and/or desk) (Can be embedded within encrypted data section.)	String

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
57	TargetSubID	N	"ADMIN" reserved for administrative messages not intended for a specific user. Assigned value used to identify specific individual or unit intended to receive the message.	String
143	TargetLocationID	N	Trading partner LocationID (i.e. geographic location and/or desk) (Can be embedded within encrypted data section.)	String
129	DeliverToSubID	N	Trading partner SubID used when delivering messages via a third party. (Can be embedded within encrypted data section). Not supported.	String
145	DeliverToLocationID	N	Trading partner LocationID (i.e. geographic location and/or desk) used when delivering messages via a third party. (Can be embedded within encrypted data section). Not supported.	String
43	PossDupFlag	N	Always required for retransmitted messages, whether prompted by the sending system or as the result of a resend request. (Can be embedded within encrypted data section.)	Boolean
97	PossResend	N	Required when message may be duplicate of another message sent under a different sequence number. (Can be embedded within encrypted data section.)	Boolean
52	SendingTime	Y	Can be embedded within encrypted data section.	UTCTimeStamp
122	OrigSendingTime	N	Required for message resent as a result of a ResendRequest. If data is not available set to same value as SendingTime (can be embedded within encrypted data section.)	UTCTimeStamp
347	MessageEncoding	N	Type of message encoding (non-ASCII (non-English) characters) used in a message's "Encoded" fields.	String
369	LastMsgSeqNumProcessed	N	Not supported	SeqNum

A.2 Standard Trailer

Each message, administrative or application is terminated by a standard trailer. The trailer is used to segregate messages and contains the three digit character representation of the Checksum value.

The standard message trailer format is as follows.

Table 51 – Standard Message Trailer

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
10	CheckSum	Y	(Always unencrypted, always last field in message)	String

Appendix B - Component Blocks

B.1 Instrument (symbology) Component Block

The Instrument component block contains all the fields commonly used to describe a security or instrument. Typically the data elements in this component block are considered the static data of a security which may be commonly found in a security master database (reference database). The Instrument component block can be used to describe any asset type supported by FIX.

The Instrument component, when part of a transaction that is inbound to the Exchange can only contain the following fields:

- SecurityID (48)
- SecurityIDSource (22)
- SecuritySubType (762)

The SecurityStatus, SecurityList and SecurityDefinition responses will return the following tags: 22, 48, 55, 106, 107, 167, 223, 224, 225, 762 and 1227. SecurityList and SecurityDefinition will additionally return the following: 201, 202, 454, 455, 456, 541, 350, 351, 1359, 1360.

TradeCaptureReport will return the following tags: 22, 48, 541 and 762.

All other messages referencing security information will contain fields 22, 48 and 762 only.

Table 52 – Instrument Component Block

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
22	SecurityIDSource	N	Identifies class or source of the SecurityID (48) value. Required if SecurityID is specified.	String
48	SecurityID	N	Unique marketplace assigned identifier number for an order book. Required for inbound transactions to the Exchange except for OrderCancelReplaceRequest(G), OrderCancelRequest(F) and Order Status Request(H).	String
55	Symbol	N	Ticker symbol or human readable representation of the security. In X-stream, this is the SecShortName.	String
454	NoSecurityAltID	N	Number of lternate Security Identifies. Always 1 if presented	NumInGroup
455	SecuirtyAltID	N	ISIN Code	String
456	SecurityAltIDSource	N	4 – ISIN number	String
762	SecuritySubType	N	In X-stream, this field is used to specify board on which SecurityID is listed. This fiels is equivelant to MarketSegmentID (1300) in Security List (y) and Security Definition (d)	String
106	Issuer	N	Issuer of security	String
107	SecurityDesc	N	Optional textual description of the security.	String
167	SecurityType	N	Indicates type of security. Valid values:	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			CS = Common Stock FUT = Future OPT = Option NONE = No Security Type	
223	CouponRate	N	For Fixed Income.	Percentage
224	CouponPaymentDate	N	Date interest is to be paid. Used in identifying Corporate Bond issues.	LocalMktDate
225	IssueDate	N	The date when a bond or stock offering is issued.	LocalMktDate
541	MaturityDate	N	Specifies the maturity date or expiry date of a option.	LocalMktDate
202	StrikePrice	N	Strike Price for an Option.	Price
201	PutOrCall	N	Indicates whether an option contract is a put or call	Int
350	EncodedSecurityDescLen	N	Must be set if EncodedSecurityDesc field is specified and must immediately precede it.	Length
351	EncodedSecurityDesc	N	Encoded (non-ASCII characters) representation of the SecurityDesc field in the encoded format specified via the MessageEncoding field.	Data
1359	EncodedSymbolLen	N	*** NSE Specific extension *** Byte length of encoded (non-ASCII characters) EncodedSymbol(1360) field.	Length
1360	EncodedSymbol	N	*** NSE Specific extension *** Encoded (non-ASCII characters) representation of the Symbol(55) field in the encoded format specified via the MessageEncoding(347) field. If used, the ASCII (English) representation can also be specified in the Symbol field.	Data
1227	ProductComplex	N	Identifier for sector.	String

B.2 Parties Component Block

The Parties component is used to provide identifiers for parties involved in the transaction (e.g. firm, trader, Exchange, etc.).

The Parties component block is used to identify and convey information on the entities both central and peripheral to the financial transaction represented by the FIX message containing the Parties Block. The Parties block allows many different types of entities to be expressed through use of the PartyRole field and identifies the source of the PartyID through the PartyIDSource. Entities can encompass:

- EnteringTrader
- EnteringFirm
- ContraTrader
- ContraFirm
- ExecutingTrader

- ExecutingFirm
- OrderOriginationTrader

Table 53 – Investment Firm Parties Component Block

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
453	NoPartyIDs		N	Repeating group below should contain unique combinations of PartyID, PartyIDSource, and PartyRole	NumInGrp
→	448	PartyID	N	Used to identify source of PartyID. Required if PartyIDSource is specified. Required if NoPartyIDs > 0.	String
→	447	PartyIDSource	N	Used to identify class source of PartyID value. Required if PartyID is specified. Required if NoPartyIDs > 0.	Char
→	452	PartyRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0.	Int

B.2.1 Examples

Firm and individual User for whom the transaction applies:

- Broker Firm

  PartyID = "…" – the identifier of the firm

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "1" – Executing Firm

- User

  PartyID = "…" – the identifier of the user

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "12" – Executing Trader

In cases the transaction is entered on behalf of the real owner and the marketplace validates authorization in those cases:

- Broker Firm

  PartyID = "…" – the identifier of the firm on behalf of the real owner

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "7" – Entering Firm

- User

  PartyID = "…" – the identifier of the user on behalf of the real owner

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "36" – Entering Trader

B.3 YieldData Component Block

The YieldData component block conveys yield information for a given Fixed Income security.

Table 54 – YieldData Component Block

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
235	YieldType	N	Type of yield.	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
236	Yield	N	Yield percentage	Percentage
696	YieldRedemptionDate	N	Date to which the yield has been calculated (i.e. maturity, par call or current call, pre-refunded date).	LocalMktDate
697	YieldRedemptionPrice	N	Price to which the yield has been calculated.	Price

B.4 TriggeringInstruction Component Block

The TriggeringInstruction component block specifies the conditions under which an order will be triggered by market events as well as behaviour of the order in the market once it is triggered.

Note: Orders with triggers will not be visible in the order book until the TriggerType event occurs. The OrdStatus (39) field in the Execution Report will return 'X' – Order with trigger in the book but not active, e.g. Order has not been triggered.

Triggered orders when activated (e.g. when the TriggerType occurs) if not immediately traded will cause an unsolicited Execution Report to be sent to the order initiator indicating that the order has become active and available for trading in the order book. If the order is immediately traded (partially or completely) a normal trade execution report will be returned.

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
1100	TriggerType	N	Defines when the trigger will hit, i.e. the action specified by the trigger instructions will come into effect. Valid Values: 4 = Price Movement	Char
1101	TriggerAction	N	Defines the type of action to take when the trigger hits. Valid Values: 1 = Activate	Char
1102	TriggerPrice	N	The price at which the trigger should hit.	Price
1107	TriggerPriceType	N	The type of price that the trigger is compared to. Valid Values: 1 = Best Offer 2 = Last Trade 3 = Best Bid	Char
1109	TriggerPriceDirection	N	The side from which the trigger price is reached. Valid Values: U= Trigger if the price of the specified type goes UP to or through the specified Trigger Price. D= Trigger if the price of the specified type goes DOWN to or through the specified Trigger Price.	Char

B.5 LinesofTextGroup Component Block

The LinesOfTextGroup component block is used to provide arbitrary text and non-printable information.

Table 55 – LinesofTextGroup Component Block

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
Start of Component block, expanded in line < LinesOfTextGroup >					
33	NoLinesOfText		Y	Specifies the number of repeating lines of text.	NumInGrp
→	58	Text	Y	Free format text string	String
→	354	EncodedTextLen	N	Must be set if EncodedText field is specified and must immediately precede it. Byte Length of encoded (non-ASCII) characters.	Length
→	355	EncodedText	N	Encoded (non-ASCII characters) representation of the Text field in the encoded format specified via the MessageEncoding.	Data
End of Component block, expanded in line < LinesOfTextGroup >					

Appendix C - Field Enumerations

C.1 Field Enumerations Sorted by Tag Value

Table 56 – Field Enumerations Sorted by Tag Value

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
18	ExecInst	N	Instructions for order handling. Valid values: 'G' – All or None (AON). 'w' – Whole or None (WON)	Char
22	SecurityIDSource	N	Identifies class or source of the SecurityID (48) value. Valid values: 99 – Marketplace assigned identifier	String
39	OrdStatus	Y	Describes the current state of an order. Valid values are: 0 – New 1 – Partially filled 2 – Filled 4 – Cancelled 5 – Replaced 8 – Rejected 9 – Suspended (Not Supported) C – Expired *** NSE Defined *** U – Order is Unplaced X – Order with trigger in the book but not active (e.g. Order has not been triggered). Z – Private Order	Char
40	OrdType	Y	Indicates the type of order. Valid values are: 1 – Market – The Price (44) field is not used, the order executes against the best prices order on the opposite side. 2 – Limit – The Price (44) field is specified and the order will execute at this price or better. 3 – Stop/Stop Loss – A type of market order that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). The Price (44) field is not specified, but the TriggerPrice (1102) is. The order will be activated as a Market order when the TriggerPrice is reached. 4 – Stop Limit – A type of limit order	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). Specifies both the Price (44) and the TriggerPrice (1102) field. The order will be activated as a Limit order (using the specified Price as the limit price) when the TriggerPrice is reached.	
54	Side	Y	Optional qualifier used to indicate the side of the market. Valid values are: 1 – Buy 2 – Sell	Char
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Valid values are: 0 – Day 1 – Good till cancelled 3 – Immediate or Cancel (IOC) 4 – Fill or Kill (FoK) 6 – Good till date (GTD) 7 – Pre Close S – Session Orders	Char
61	Urgency	N	Urgency Flag. Valid values are: 0 – Normal 1 – Flash 2 – Background	Char
102	CxlRejReason	N	Identifies the reason for the cancel rejection. Valid values: 1 – Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) '99' – Other. Refer to returned Text (58) field for exact reason for rejection.	Int
103	OrdRejReason	N	For optional use with ExecType = 8 (Rejected). Code to identify reason for order rejection. Valid values: 5 = Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) 99 – Other. Refer to returned Text (58) field for exact reason for rejection.	Int
127	DKReason	Y	Reason for execution rejection. Valid values: A – Unknown Symbol B – Wrong Side C – Quantity Exceeds Order	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			D – No Matching Order E – Price Exceeds Limit F – Calculation Difference Z – Other	
150	ExecType	Y	Type of Execution being reported. Describes the specific ExecutionRpt (i.e. Pending Cancel) while OrdStatus (39) will always identify the current order status (i.e. Partially Filled) Valid values: 0 – New 3 – Done for day 4 – Cancelled 5 – Replaced 6 – Pending Cancel (e.g. result of Order Cancel Request) 7 – Stopped 8 – Rejected 9 – Suspended C – Expired F – Trade (partial fill or fill) H – Trade Cancel I – Order Status U – Order is Unplaced (NSE Only)	Char
235	YieldType	N	Supported values are: OPENAVG – Open Average Yield CLOSE – Closing Yield TRUE High – Trading session high yield Low – Trading session low yield Last – Last yield WAvg – Weighted Average Change – Change from reference yield.	String
263	SubscriptionRequestType	N	Used to subscribe for Quote Status Report messages. Subscribe or unsubscribe for security status to security specified in request. Subscription type request. Valid values are: 0 – Snapshot 1 – Snapshot+Updates (Subscribe) 2 – Disable previous Snapshot+Update Request (unsubscribe)	Char
264	MarketDepth	Y	Depth of market for Book Snapshot / Incremental updates. Valid values: 0 – full book depth 1 – top of book 2 and above – book depth (number of levels). Note: For market by price (MBP) this is limited to a depth of 100. For market	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			by order (MBO) or non-aggregated book, this is limited to 100.	
265	MDUpdateType	N	Required if SubscriptionRequestType = Snapshot + Updates (1). Specifies the type of Market Data update. Valid values: 1 – Incremental refresh	Int
266	AggregatedBook	N	Specifies whether or not book entries should be aggregated. Valid values: Y – book entries to be aggregated N – book entries should not be aggregated	Boolean
269	MDEntryType	Y	Must be first field in 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. For requests the following are valid values: 0 - Order information – requests all order related information in market by order and market by price messages (e.g. bids, offers, etc.). 2 - Trade information – returns all trade information and statistics. 3 - Index information – returns all index related information. 'a' - Security Statistics (NSE specific) – returns security specific market statistics. For responses to market data requests the following are valid values: Valid values: 0 – Bid 1 – Offer 2 – Trade 3 – Index Value 4 – Opening Price 5 – Closing Price 6 – Settlement Price 7 – Trading Session High Price 8 – Trading Session Low Price 9 – Trading Session VWAP Price C – Open Interest J – Empty Book j - No Trades Exist A - Imbalance For 2, 3, 4, 5, 7, 8 the MarketDepth (264) must be set to '1' = 'Top of Book'. *** NSE Extensions ***	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			s – Unadjusted Previous Closing price t – Delta change from reference price u – Previous closing price v – Total value of security traded today w – Total number of shares traded today (returned in tag 1020) x – Reference Price (returned in tag 270) y – 52 week high and low (returned in tags 332 and 333. z – Indicative opening price and quantity (returned in tags 270, 271, 336) p – Indicative closing price and quantity (returned in tags 270, 271, 336)	
274	TickDirection	N	Direction of the "tick". Valid values: 0 – Plus Tick 1 – Zero-Plus Tick 2 – Minus Tick 3 – Zero-Minus Tick	Char
279	MDUpdateAction	Y	Must be first field in this repeating group. Valid values: 0 – New 1 – Change 2 – Delete	Char
281	MDReqRejReason	N	Reason for the rejection of a Market Data request. Valid values: 0 – Unknown symbol 1 – Duplicate MDReqID 2 – Insufficient Bandwidth 3 – Insufficient Permissions 4 – Unsupported Subscription Request Type 5 – Unsupported MarketDepth 6 – Unsupported MDUpdateType 8 – Unsupported MDEntryType 9 – Unsupported TradingSessionID	Char
292	CorporateAction	N	Identifies the type of Corporate Action that triggered the update. Also referred to as 'Basis of Quotation'. Valid values: A – Ex-Dividend B – Ex-Distribution C – Ex-Rights D – New E – Ex-Interest F – Cash Dividend G – Stock Dividend (Cum Divided)	MultipleCharValue

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			H – Stock Split I – Reverse Stock Split L – Liquidation Reorganization M – Merger Reorganization N – Rights Offering (Cum Rights) R – Warrant *** NSE Extensions *** a – Cum Bonus b – Cum Demerge c – Cum Interest d – Cum Listing e – Cum Right of Conversion f – Call Paid g – Cum Delisting h – Offer Closing i – Unlisted j – Ex Bonus k – Ex Demerge l – Ex Listing m – Ex Merge n – Ex Right of Conversion o – Ex Split p – Ex Delisting	
321	SecurityRequestType	N	Type of Security Definition Request. Valid values: 4 – Symbol (security ID in X-stream) 8 – All Securities 9 – MarketID	Int
323	SecurityResponseType	N	Type of Security Definition response. Valid values: 4 – List of securities returned per request 6 – Cannot match selection criteria	Int
326	SecurityTradingStatus	N	Identifies the trading status applicable to the transaction. Valid values: 1 – Opening delay 2 – Trading halt 17 – Ready to trade 18 – Not available for trading 19 – Not traded on this market 20 – Unknown or Invalid	Int
336	TradingSessionID	Y	Identifier for Trading Session. A trading session spans an extended period of time that can also be expressed informally in terms of the trading day. Usage is determined by market or counterparties. For trading session related messages TradingSessionID will map to the current trading session state (e.g. pre-open, open, closed, etc.)	String
340	TradSesStatus	Y	State of trading session. Valid values	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			are: 6 – Request rejected *** NSE Specific *** 100 – Pending – Indicates that trading session has not been started 101 – Triggered – Indicates trading session has either occurred or is the current session 102 – Deleted – This trading session has been removed from the trading schedule.	
380	BusinessRejectReason	Y	Valid values: 0 – Other 1 – Unknown ID 2 – Unknown Security 3 – Unknown Message Type 4 – Application not available 5 – Conditionally required field missing 6 – Not Authorized	Int
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to. Valid values are: 1 – Order Cancel Request 2 – Order Cancel/Replace Request	Char
447	PartyIDSource	N	Used to identify class source of PartyID value. Required if PartyID is specified. Required if NoPartyIDs > 0. Valid values are: C – Participant identifier	Char
452	PartyRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0. Valid values are: 1 – Executing Firm 3 – Client ID 4 – Clearing Firm 5 – Investor ID 7 – Entering Firm 11 – Order origination trader 12 – Executing Trader 13 – Order origination firm 17 – Contra Firm 22 – Exchange 32 - Beneficiary 36 – Entering trader 37 – Contra trader 67 – Investment Firm 73 – Execution venue	Int
559	SecurityListRequestType	N	Specifies the criteria of the request: 0 – Symbol 4 – All Securities 5 – MarketID (Specific Market)	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
560	SecurityRequestResult	N	Result of the Security Request identified by the SecurityReqID. Valid values: 0 – Valid request 1 – Invalid or unsupported request 2 – No instruments found that match selection criteria 3 – Not authorized to retrieve instrument data 4 – Instrument data temporarily unavailable 5 – Request for instrument data not supported *** 100+ NSE Specific*** 100 – Invalid MarketID	Int
567	TradSesStatusRejReason	N	Used with TradSesStatus = "Request Rejected". Valid values are: 1 – Unknown trading session id	Int
569	TradeRequestType	Y	Type of Trade Capture Report. Valid values: 0 – All Trades	
574	MatchType	N	The point in the matching process at which this trade was matched. Valid values: 1 - One-Party Trade Report (privately negotiated trade) 2 - Two-Party Trade Report (privately negotiated trade) 4 - Auto-match	String
980	SecurityUpdateAction	N	Specifies the update action for the security. Valid values: A – Add D – Delete M – Modify	Char
1021	MDBookType	N	Describes the type of book for which the feed is intended. Used when multiple feeds are provided over the same connection. Valid Values: 1 = Top of Book 2 = Price Depth 3 = Order Depth	Int
1057	AggressorIndicator	N	Used to identify whether the order initiator is an aggressor or not in the trade. Valid values: Y – Order initiator is aggressor N – Order initiator is passive	Boolean
1093	LotType	N	Used to indicate a Block Divestment – only valid value is '3' Block Lot	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
1174	SecurityTradingEvent	N	Identifies an event related to a SecurityTradingStatus(326) Valid values are: 3 – Price Volatility Interruption	Char
1324	ListUpdateAction	N	Specifies the action for a security list. If provided then the Instrument occurrence has explicitly changed: Valid values are: A – Add D – Delete M - Modify	Char
1327	TradSesUpdateAction	N	Specifies the action taken for the specified trading sessions. Valid values: A - Add D - Delete M - Modify	Char
1369	MassActionReportID	Y	Unique Identifier for the Order Mass Action Report. This is a V5.0 tag value.	String
1373	MassActionType	Y	Specifies the type of mass action requested. Valid values: 3 - Cancel orders *** NSE Specific *** 100 – Order Status	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. Valid values: 1 - All orders for a security 7 - All orders 8 - All orders for a Market	Int
1375	MassActionResponse	Y	Indicates the action taken by the counterparty order handling system as a result of the Action Request 0 - Request rejected. 1 - Accepted	Int
1376	MassActionRejectReason	N	Indicates why Order Mass Action Request was rejected. Required if MassActionResponse = 0. Valid values: 0 - Mass Action Not Supported 1 - Invalid or unknown security 7 - Invalid or unknown Market 8 – Invalid or unknown Market Segment 99 - Other	Int
1395	MarketUpdateAction	N	Specifies the action taken for the specified MarketID(1301) + MarketSegmentID(1300).	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Valid values are: A = Add D = Delete M =Modify	

C.2 Field Enumerations Sorted By Tag Name

Table 57 – Field Enumerations Sorted By Tag Name

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
266	AggregatedBook	N	Specifies whether or not book entries should be aggregated. Valid values: Y – book entries to be aggregated N – book entries should not be aggregated	Boolean
1057	AggressorIndicator	N	Used to identify whether the order initiator is an aggressor or not in the trade. Valid values: Y – Order initiator is aggressor N – Order initiator is passive	Boolean
380	BusinessRejectReason	Y	Valid values: 0 – Other 1 – Unknown ID 2 – Unknown Security 3 – Unknown Message Type 4 – Application not available 5 – Conditionally required field missing 6 – Not Authorized	Int
292	CorporateAction	N	Identifies the type of Corporate Action that triggered the update. Also referred to as 'Basis of Quotation'. Valid values: A – Ex-Dividend B – Ex-Distribution C – Ex-Rights D – New E – Ex-Interest F – Cash Dividend G – Stock Dividend (Cum Divided) H – Stock Split I – Reverse Stock Split L – Liquidation Reorganization M – Merger Reorganization N – Rights Offering (Cum Rights) R – Warrant *** NSE Specific *** a – Cum Bonus b – Cum Demerge c – Cum Interest d – Cum Listing e – Cum Right of Conversion f – Call Paid g – Cum Delisting h – Offer Closing i – Unlisted j – Ex Bonus	MultipleCharValue

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			k – Ex Demerge l – Ex Listing m – Ex Merge n – Ex Right of Conversion o – Ex Split p – Ex Delisting	
102	CxlRejReason	N	Identifies the reason for the cancel rejection. Valid values: 1 – Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) '99' – Other. Refer to returned Text (58) field for exact reason for rejection.	Int
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to. Valid values are: 1 – Order Cancel Request 2 – Order Cancel/Replace Request	Char
127	DKReason	Y	Reason for execution rejection. Valid values: A – Unknown Symbol B – Wrong Side C – Quantity Exceeds Order D – No Matching Order E – Price Exceeds Limit F – Calculation Difference Z – Other	Char
18	ExecInst	N	Instructions for order handling. Valid values: 'G' – All or None (AON). 'w' – Whole or None (WON)	Char
150	ExecType	Y	Type of Execution being reported. Describes the specific ExecutionRpt (i.e. Pending Cancel) while OrdStatus (39) will always identify the current order status (i.e. Partially Filled) Valid values: 0 – New 3 – Done for day 4 – Cancelled 5 – Replaced 6 – Pending Cancel (e.g. result of Order Cancel Request) 7 – Stopped 8 – Rejected 9 – Suspended C – Expired F – Trade (partial fill or fill) H – Trade Cancel I – Order Status U – Order is Unplaced	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
1324	ListUpdateAction	N	Specifies the action for a security list. If provided then the Instrument occurrence has explicitly changed: Valid values are: A – Add D – Delete M - Modify	Char
1093	LotType	N	Used to indicate a Block Divestment – only valid value is '3' Block Lot	Char
264	MarketDepth	Y	Depth of market for Book Snapshot / Incremental updates. Valid values: 0 – full book depth 1 – top of book 2 and above – book depth (number of levels) Note: For market by price (MBP) this is limited to a depth of 100. For market by order (MBO) or non-aggregated book, this is limited to 100.	Int
1395	MarketUpdateAction	N	Specifies the action taken for the specified MarketID(1301) + MarketSegmentID(1300). Valid values are: A = Add D = Delete M =Modify	Char
1369	MassActionReportID	Y	Unique Identifier for the Order Mass Action Report. This is a V5.0 tag value.	String
1373	MassActionType	Y	Specifies the type of mass action requested. Valid values: 3 - Cancel orders *** NSE Specific *** 100 – Order Status	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. Valid values: 1 - All orders for a security 7 - All orders 8 - All orders for a Market	Int
1375	MassActionResponse	Y	Indicates the action taken by the counterparty order handling system as a result of the Action Request 0 - Request rejected. 1 - Accepted	Int
1376	MassActionRejectReason	N	Indicates why Order Mass Action Request was rejected. Required if	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			MassActionResponse = 0. Valid values: 0 - Mass Action Not Supported 1 - Invalid or unknown security 8 – Invalid or unknown Market Segment 7 - Invalid or unknown Market 99 - Other	
574	MatchType	N	The point in the matching process at which this trade was matched. Valid values: 1 - One-Party Trade Report (privately negotiated trade) 2 - Two-Party Trade Report (privately negotiated trade) 4 - Auto-match	String
269	MDEntryType	Y	Must be first field in 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. For market data requests the following are valid values: 0 - Order information – requests all order related information in market by order and market by price messages (e.g. bids, offers, price, quantity, depth, etc.). 2 - Trade information – returns all trade information and statistics. 3 - Index information – returns all index related information. 'a' - Security Statistics (NSE specific) – returns security specific market statistics. For responses to market data requests the following are valid values: Valid values: 0 – Bid 1 – Offer 2 – Trade 3 – Index Value 4 – Opening Price 5 – Closing Price 6 – Settlement Price 7 – Trading Session High Price 8 – Trading Session Low Price 9 – Trading Session VWAP Price C – Open Interest J – Empty Book j - No Trades Exist	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			A - Imbalance For 2, 3, 4, 5, 7, 8 the MarketDepth (264) must be set to '1' = 'Top of Book'. *** NSE Extensions *** s – Unadjusted Previous Closing price t – Delta change from reference price u – Previous closing price v – Total value of security traded today w – Total number of shares traded today (returned in tag 1020) x – Reference Price (returned in tag 270) y – 52 week high and low (returned in tags 332 and 333. z – Indicative opening price and quantity (returned in tags 270, 271, 336) p – Indicative closing price and quantity (returned in tags 270, 271, 336)	
281	MDReqRejReason	N	Reason for the rejection of a Market Data request. Valid values: 0 – Unknown symbol 1 – Duplicate MDReqID 2 – Insufficient Bandwidth 3 – Insufficient Permissions 4 – Unsupported Subscription Request Type 5 – Unsupported MarketDepth 6 – Unsupported MDUpdateType 8 – Unsupported MDEntryType 9 – Unsupported TradingSessionID	Char
279	MDUpdateAction	Y	Must be first field in this repeating group. Valid values: 0 – New 1 – Change 2 – Delete 3 - Delete Thru 4 - Delete From	Char
265	MDUpdateType	N	Required if SubscriptionRequestType = Snapshot + Updates (1). Specifies the type of Market Data update. Valid values: 1 – Incremental refresh	Int
103	OrdRejReason	N	For optional use with ExecType = 8 (Rejected). Code to identify reason for order rejection.	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Valid values are: 5 = Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) 99 – Other. Refer to returned Text (58) field for exact reason for rejection.	
39	OrdStatus	Y	Describes the current state of an order. Valid values are: 0 – New 1 – Partially filled 2 – Filled 4 – Cancelled 5 – Replaced 8 – Rejected 9 - Suspended A – Pending New (used only for vetting by X-stream). C – Expired *** NSE Defined *** U – Order is Unplaced X – Order with trigger in the book but not active (e.g. Order has not been triggered). Z – Private Order	Char
40	OrdType	Y	Indicates the type of order. Valid values are: 1 – Market – The Price (44) field is not used, the order executes against the best prices order on the opposite side. 2 – Limit – The Price (44) field is specified and the order will execute at this price or better. 3 – Stop/Stop Loss – A type of market order that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). The Price (44) field is not specified, but the TriggerPrice (1102) is. The order will be activated as a Market order when the TriggerPrice is reached. 4 – Stop Limit – A type of limit order that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). Specifies both the Price (44) and the TriggerPrice (1102) field. The order will be activated as a Limit order (using the specified Price as the limit price) when the TriggerPrice is reached.	Char
447	PartyIDSource	N	Used to identify class source of PartyID value. Required if PartyID is specified.	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Required if NoPartyIDs > 0. Valid values are: C – Participant identifier	
452	PartyRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0. Valid values are: 1 – Executing Firm 3 – Client ID 4 – Clearing Firm 5 – Investor ID 7 – Entering Firm 11 – Order origination trader 12 – Executing Trader 13 – Order origination firm 17 – Contra Firm 22 – Exchange 32 - Beneficiary 36 – Entering trader 37 – Contra trader 67 – Investment Firm 73 – Execution venue	Int
22	SecurityIDSource	N	Identifies class or source of the SecurityID (48) value. Valid values: 99 – Marketplace assigned identifier	String
559	SecurityListRequestType	N	Specifies the criteria of the request: 0 – Symbol 4 – All Securities 5 – MarketID (Specific Market)	Int
560	SecurityRequestResult	N	Result of the Security Request identified by the SecurityReqID. Valid values: 0 – Valid request 1 – Invalid or unsupported request 2 – No instruments found that match selection criteria 3 – Not authorized to retrieve instrument data 4 – Instrument data temporarily unavailable 5 – Request for instrument data not supported *** 100+ NSE Specific*** 100 – Invalid MarketID	Int
321	SecurityRequestType	N	Type of Security Definition Request. Valid values: 4 – Symbol (security ID in X-stream) 8 – All Securities 9 – MarketID	Int
323	SecurityResponseType	N	Type of Security Definition response. Valid values:	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			4 – List of securities returned per request 6 – Cannot match selection criteria	
1174	SecurityTradingEvent	N	Identifies an event related to a SecurityTradingStatus(326) Valid values are: 3 – Price Volatility Interruption	Char
326	SecurityTradingStatus	N	Identifies the trading status applicable to the transaction. Valid values: 1 – Opening delay 2 – Trading halt 17 – Ready to trade 18 – Not available for trading 19 – Not traded on this market 20 – Unknown or Invalid	Int
980	SecurityUpdateAction	N	Specifies the update action for the security. Valid values: A – Add D – Delete M – Modify	Char
54	Side	Y	Optional qualifier used to indicate the side of the market. Valid values are: 1 – Buy 2 – Sell	Char
263	SubscriptionRequestType	N	Used to subscribe for Quote Status Report messages. Subscribe or unsubscribe for security status to security specified in request. Subscription type request. Valid values are: 0 – Snapshot 1 – Snapshot+Updates (Subscribe) 2 – Disable previous Snapshot+Update Request (unsubscribe)	Char
274	TickDirection	N	Direction of the "tick". Valid values: 0 – Plus Tick 1 – Zero-Plus Tick 2 – Minus Tick 3 – Zero-Minus Tick	Char
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Valid values are: 0 – Day 1 – Good till cancelled 3 – Immediate or Cancel (IOC) 4 – Fill or Kill (FoK) 6 – Good till date (GTD) 7 – Pre Close	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			S - Session Orders	
569	TradeRequestType	Y	Type of Trade Capture Report. Valid values: 0 – All Trades	569
336	TradingSessionID	Y	Identifier for Trading Session. A trading session spans an extended period of time that can also be expressed informally in terms of the trading day. Usage is determined by market or counterparties. For trading session related messages TradingSessionID will map to the current trading session state (e.g. pre-open, open, closed, etc.)	String
340	TradSesStatus	Y	State of trading session. Valid values are: 6 – Request rejected *** NSE Specific *** 100 – Pending – Indicates that trading session has not been started 101 – Triggered – Indicates trading session has either occurred or is the current session 102 – Deleted – This trading session has been removed from the trading schedule.	Int
567	TradSesStatusRejReason	N	Used with TradSesStatus = "Request Rejected". Valid values are: 1 – Unknown trading session id	Int
1327	TradSesUpdateAction	N	Specifies the action taken for the specified trading sessions. Valid values: A - Add D - Delete M - Modify	Char
61	Urgency	N	Urgency Flag. Valid values are: 0 – Normal 1 – Flash 2 – Background	Char
235	YieldType	N	Supported values are: OPENAVG – Open Average Yield CLOSE – Closing Yield TRUE High – Trading session high yield Low – Trading session low yield Last – Last yield WAvg – Weighted Average Change – Change from reference yield.	String

Appendix D - FIX Data Types

Data types (with the exception of those of type "data") are mapped to ASCII strings as follows.

int	Sequence of digits without commas or decimals and optional sign character (ASCII characters "-" and "0" - "9"). The sign character utilizes one byte (i.e. positive int is "99999" while negative int is "-99999"). Note that int values may contain leading zeros (e.g. "00023" = "23"). Examples: 723 in field 21 would be mapped int as |21=723|. -723 in field 12 would be mapped int as |12=-723| The following data types are based on int.	
	Length	int field representing the length in bytes. Value must be positive.
	TagNum	int field representing a field's tag number when using FIX "Tag=Value" syntax. Value must be positive and may not contain leading zeros.
	SeqNum	int field representing a message sequence number. Value must be positive.
	NumInGroup	int field representing the number of entries in a repeating group. Value must be positive.
	DayOfMonth	int field representing a day during a particular month (values 1 to 31).
float	Sequence of digits with optional decimal point and sign character (ASCII characters "-", "0" - "9" and "."); the absence of the decimal point within the string will be interpreted as the float representation of an integer value. All float fields must accommodate up to fifteen significant digits. The number of decimal places used should be a factor of business/market needs and mutual agreement between counterparties. Note that float values may contain leading zeros (e.g. "00023.23" = "23.23") and may contain or omit trailing zeros after the decimal point (e.g. "23.0" = "23.0000" = "23" = "23."). Note that fields which are derived from float may contain negative values unless explicitly specified otherwise. The following data types are based on float.	
	Qty	float field capable of storing either a whole number (no decimal places) of "shares" (securities denominated in whole units) or a decimal value containing decimal places for non-share quantity asset classes (securities denominated in fractional units).
	Price	float field representing a price. Note the number of decimal places may vary. For certain asset classes, prices may be negative values. For example, prices for options strategies can be negative under certain market conditions (see FIX Specifications Volume 7: FIX Usage by Product for asset classes that support negative price values).
	PriceOffset	float field representing a price offset, which can be mathematically added to a "Price". Note the number of decimal places may vary and some fields such as LastForwardPoints may be negative.
	Amt	float field typically representing a Price times a Qty
	Percentage	float field representing a percentage (e.g. 0.05 represents 5% and 0.9525 represents 95.25%). Note the number of decimal places may vary.
char	Single character value, can include any alphanumeric character or punctuation except the delimiter. All char fields are case sensitive (i.e. m != M). The following fields are based on char.	

	Boolean	char field containing one of two values: 'Y' = True/Yes 'N' = False/No
String	Alpha-numeric free format strings, can include any character or punctuation except the delimiter. All String fields are case sensitive (i.e. morstatt != Morstatt).	
	MultipleCharValue	string field containing one or more space delimited single character values (e.g. |18=2 A F|).
	MultipleStringValue	string field containing one or more space delimited multiple character values (e.g. |277=AV AN A|).
	Country	string field representing a country using ISO 3166 Country code (2 character) values (see FIX Specifications Volume 6 - Appendix 6-B).
	Currency	string field representing a currency type using ISO 4217 Currency code (3 character) values (see FIX Specifications Volume 6 - Appendix 6-A).
	Exchange	string field representing a market or exchange using ISO 10383 Market Identifier Code (MIC) values (see FIX Specifications Volume 6 - Appendix 6-C).
	MonthYear	string field representing month of a year. An optional day of the month can be appended or an optional week code. Valid formats: YYYYMM YYYYMMDD YYYYMMWW Valid values: YYYY = 0000-9999; MM = 01-12; DD = 01-31; WW = w1, w2, w3, w4, w5.
	UTCTimestamp	string field representing Time/date combination represented in UTC (Universal Time Coordinated, also known as "GMT") in either YYYYMMDD-HH:MM:SS (whole seconds) or YYYYMMDD-HH:MM:SS.sss (milliseconds) format, colons, dash, and period required. Valid values: * YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second) (without milliseconds). * YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds). Leap Seconds: Note that UTC includes corrections for leap seconds, which are inserted to account for slowing of the rotation of the earth. Leap second insertion is declared by the International Earth Rotation Service (IERS) and has, since 1972, only occurred on the night of Dec. 31 or Jun 30. The IERS considers March 31 and September 30 as secondary dates for leap second insertion, but has never utilized these dates. During a leap second insertion, a UTCTimestamp field may read "19981231-23:59:59", "19981231-23:59:60", "19990101-00:00:00". (see http://tycho.usno.navy.mil/leapsec.html)
	UTCTimeOnly	string field representing Time-only represented in UTC (Universal Time Coordinated, also known as "GMT") in either HH:MM:SS (whole seconds) or HH:MM:SS.sss (milliseconds) format, colons, and period required. This

	special-purpose field is paired with UTCDateOnly to form a proper UTCTimestamp for bandwidth-sensitive messages. Valid values: HH = 00-23, MM = 00-60 (60 only if UTC leap second), SS = 00-59. (without milliseconds) HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds).
UTCDateOnly	string field representing Date represented in UTC (Universal Time Coordinated, also known as "GMT") in YYYYMMDD format. This special-purpose field is paired with UTCTimeOnly to form a proper UTCTimestamp for bandwidth-sensitive messages. Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31.
LocalMktDate	string field representing a Date of Local Market (as opposed to UTC) in YYYYMMDD format. This is the "normal" date field used by the FIX Protocol. Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31.
data	string field containing raw data with no format or content restrictions. Data fields are always immediately preceded by a length field. The length field should specify the number of bytes of the value of the data field (up to but not including the terminating SOH). **Caution**: The value of one of these fields may contain the delimiter (SOH) character. Note that the value specified for this field should be followed by the delimiter (SOH) character as all fields are terminated with an "SOH".