Nasdaq

Genium INET^{SM}

FIX for BIST

Version: 1.0.41

Document version:	1.0.41
Publication date:	2019-10-22

All content in this document is owned, or licensed, by Nasdaq, Inc. or its affiliates ('Nasdaq'). Unauthorized use is prohibited without written permission of Nasdaq. While reasonable efforts have been made to ensure that the contents of this document are accurate, the document is provided strictly "as is", and no warranties of accuracy are given concerning the contents of the information contained in this document, including any warranty that the document will be kept up to date. Nasdaq reserves the right to change details in this document without notice. To the extent permitted by law no liability (including liability to any person by reason of negligence) will be accepted by Nasdaq or its employees for any direct or indirect loss or damage caused by omissions from or inaccuracies in this document.

.

Contents

1 References

[1]
FIX 5.0 SP2 Protocol Specification
http://fixprotocol.org/specifications/fix5.0sp2spec

[2]
FIX Protocol Limited, Market Data Optimization Working Group:
Recommended Practices for Book Management, Version 2.00, 2007
http://fixprotocol.org/documents/2518/MDOWG_Book_Mgt v20.doc

2 Overview

This document contains the specification for the FIX interface to the Genium INET trading system. The interface is based on the FIX Protocol 5.0 SP2 standard (Financial Information exchange). More detailed information about the standard can be found in FIX specification document see [1].

The interface implemented by Genium INET follows the FIX specifications as far as possible. In the absolute majority of cases the structure and semantics of the messages are identical to the standard. In some cases, the protocol has been extended to cover functions not considered by the standard. These extensions are clearly detailed in the document. In other cases, the standard is ambiguous or indicates that the details should be bilaterally agreed between the parties. In these cases this manual provides a detailed description to avoid any possible ambiguity.

To avoid possible duplication in the sources of information, this document does not include explanations of those matters that comply exactly with the standard. Therefore, the standard documentation should be considered as the main source of information for any matter that is not explicitly covered in this manual.

This specification tries not to repeat what is specified in the FIX standard. In many cases however, the FIX standard is, by necessity, more generic than that required for a specific marketplace. In other cases Genium INET has found reason to clarify matters. Genium INET tries to be explicit on deviations from the FIX standard specification in order to avoid confusion.

2.1 Supported messages

2.1.1 Administrative messages

Logon
Logout
Sequence Reset
Resend Request
Reject
Heartbeat
Test Request

2.1.2 Inbound Application messages

User Request
New Order Single
Order Cancel Replace Request
Order Cancel Request
Mass Quote
Trade Capture Report
Security Definition Request

2.1.3 Outbound Application messages

User Response
User Notification
Execution Report
Order Cancel Reject
Business Message Reject
Mass Quote Acknowledgement
Trade Capture Report
Trade Capture Report Ack
Security Definition – Registration Response

.

2.2 The Genium INET Repository

This specification defines the full set of messages, fields and enumerated values that can be used. As with most FIX implementations, this only supports a small subset of all available messages, components, fields and enumerated values defined in FIX 5.0 SP2. An FPL-formatted repository corresponding to this specification is delivered separately.
NOTE: Inbound messages not conforming to this spec, will be rejected with a session-level Reject message.

3 The FIX Session

The session layer conforms to the standard FIX session. Please see the standard FIX specification for additional details.

3.1 CompIDs

The Sender- and TargetCompID uniquely define the FIX session. A session can only be active (established) between two hosts simultaneously. Any attempts to establish a second FIX session using the same CompIDs (for instance to a backup gateway) in parallel will be rejected.

- The TargetCompID for transactions sent *inbound* to the Exchange will be “BI” for production and “BI_TEST” for test systems.
- The SenderCompID for transactions sent *outbound* from the Exchange will be “BI” for production and “BI_TEST” for test systems.

3.2 SenderSubID

Each inbound business transaction must have the SenderSubID (tag 50) field set to an authenticated user. One user can be authenticated by setting the Username and Password field in the Logon message. Additional users can be authenticated using the User Request message. See chapter 4 for a description on how to authenticate additional users.

The SenderSubID on incoming transactions will be echoed back in TargetSubID (tag 57) on outbound transactions.

NOTE: On the Logon or User Request, the SenderSubID must be set to the user id the client intends to log on.

3.3 User Authentication

Each incoming business transaction must have a username set in the SenderSubID field. The user needs must be authenticated for the transaction to be accepted. A user is authenticated by setting the Username (553) and Password (554) fields in the Logon message.

3.3.1 Renewal of passwords

A new password may be set by setting the NewPassword (925) field along with the current password in the Password (554) field. This can be done either with the Logon message or the User Request message. The SessionStatus (1409) field of the Logon returned to the client can be checked to see if the new password was accepted.

3.3.2 Expired passwords

If the password has expired when a client tries to log in, the system will respond with a Logout message with SessionStatus set to 8 – Password expired. To gain access, the client must issue a new Logon message with NewPassword set (along with the expired password in Password). If the new password is not valid, the system will respond with another Logout message. SessionStatus will be set to 3 – New session password does not comply with policy. The client will be able to log in again with another new password.

3.4 Logon

At Logon, clients are identified by:

- CompIDs

- IP Address

The Logon Username and Password fields are used to authenticate the client. When the client is authenticated, the system responds with a Logon message to the client.

3.5 Heartbeat intervals

Heartbeat intervals are negotiated at Logon using the HeartBtInt (108) field. The system allows heartbeat intervals greater than 10 seconds. **Recommended heartbeat interval is 30 s.** A heartbeat interval set lower than 10 seconds will result in a Logout response.

3.6 Encryption

The system does not support encryption.

3.7 Datatypes and required fields

This specification does not change the datatype on any fields defined in the standard FIX specification. There may be places where this specification restricts the value range of a field further than specified in standard FIX. This will be clearly marked in the spec.
All fields listed in this specification that are marked as required in the standard specification, are required also in this specification. This document specifies additional fields as required. These fields are marked with a ‘Q’ in the required column of the message listings.

3.8 Character encoding

In order to support Turkish characters,, the FIX gateway will use the 8-bit standard ISO-8859-9 encoding, often called Latin-9. The lower 7 bits are compatible with the standard 7-bit ASCII character encoding.

3.9 Session lifetime

The FIX session lifetime is restricted to one trading day. The session lifetime is not ended at connectivity loss or even Logouts. The sequence numbers are reset to one each morning.

3.10 Failover and message recovery

At reconnect and Logon standard FIX message recovery is performed. All FIX sessions have at least one primary and one secondary gateway to which the session states are fully replicated. This means that regardless to which gateway a client connects, full message recovery is provided.
A client cannot have the same FIX session active towards multiple gateway instances simultaneously.

3.10.1 Order Suspension/inactivation at connection loss

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

3.11 **FIX Session Level Test Cases**

This implementation is fully compliant with the session-level test cases specified in the standard FIX 5.0 SP2 Specification, Volume 2, section “FIX Session-level Test Cases and Expected Behaviors”. The only exception is the encryption test cases.

3.12 **The Standard Header**

All FIX messages contain a Standard Header. The header contains important information such as session identifiers (CompIDs), sequence numbers and message type and length etc.

Tag num	FIX Field name	Req'd	Comment
8	BeginString	Y	Identifies beginning of new message and protocol version. ALWAYS FIRST FIELD IN MESSAGE. Valid values: FIXT.1.1
9	BodyLength	Y	Message length, in bytes, forward to the CheckSum field. ALWAYS SECOND FIELD IN MESSAGE.
35	MsgType	Y	Defines message type ALWAYS THIRD FIELD IN MESSAGE.
49	SenderCompID	Y	As specified in separate agreement
50	SenderSubID		Required on inbound transactions. Must be set to a valid authenticated user.
56	TargetCompID	Y	As specified in separate agreement
57	TargetSubID		Should not be populated on inbound transactions. Will contain the value of incoming SenderSubID on outbound transactions. In some cases, such as in unsolicited cancels, TargetSubID will not be set.
34	MsgSeqNum	Y	Integer message sequence number.
43	PossDupFlag		Indicates possible retransmission of message with this sequence number. Always required for retransmitted messages
97	PossResend		Indicates that message may contain information that has been sent under another sequence number. Required when message may be duplicate of another message sent under a different sequence number.
52	SendingTime	Y	Time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT")
122	OrigSendingTime		Original time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT"). Required for message resent as a result of a ResendRequest.

3.12.1 Possible Duplicate vs. Possible Resend

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

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

3.13 The Standard Trailer

All FIX messages end with a Standard Trailer. The trailer only includes a simple checksum field. The details on how to calculate the checksum can be found in the standard FIX specification.

Tag num	FIX Field name	Req'd	Comment
10	CheckSum	Y	

3.14 Message Details

3.14.1 How to interpret the Required (Req'd) column

A 'Y' marks the field as required in standard FIX (and of course also in this implementation). A 'Q' means that the field is required in this implementation although it is not required in standard FIX. No entry at all means the field is optional.

3.14.2 Repeating groups

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

453	NoPartyIDs				Optional repeating group only used for on behalf of transactions and account model values.
→	448	PartyID		Q	Party identifier.
→	447	PartyIDSource		Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified.
→	802	NoPartySubIDs			Number of PartySubIDs present. Only used for PartyRole=Executing Firm. Will always be 1.
→	→	523	PartySubID	Q	Sub-identifier of party. Here Exchange code of the party.
→	→	803	PartySubIDType	Q	Type of PartySubID (523) value

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

3.14.3 Logon – inbound to Marketplace

The response to a logon is either a Logon, which denotes a successful logon, or a Logout.

A client must be prepared to handle failure scenarios including (but not limited to):

A Logon attempt may fail or be rejected for several reasons. The FIX gateway will react differently depending on the kind of failure. The two different actions it may take are:

- Silently ignore the Logon.
- If authentication fails (for security reasons).

- If the wrong Sender or Target CompID is specified.
- For other reasons specified in the standard FIX specifications.
- If the FIX gateway has no connection with the back-end system.
- Respond with a Logout.
- Logon failure for other reasons than authentication/security.

The Logout response to a Logon will always contain a note on why in the Text (58) field.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. **NOTE:** Resetting the sequence numbers will result in all prior messaging being lost. Valid values: Y = Yes
108	HeartBtInt	Y	Heartbeat interval. Any value greater than 10 s is accepted. A lower value will result in a Logout response.
553	Username	Q	User name **NOTE:** Must be in CAPTIAL LETTERS.
554	Password	Q	password (unencrypted)
925	NewPassword		Specifies a new password for the FIX Logon. The new password is used for subsequent logons.
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2
	Standard Trailer	Y	

3.14.4 Logon – outbound from Marketplace

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. Will only be set as a response to an inbound Logon with this flag set. Valid values: Y = Yes
108	HeartBtInt	Y	As specified in inbound Logon. Valid range: Greater than 10 s
1409	SessionStatus	Q	Status of the FIX session. Valid values: 0 = Session Active 1 = Session password changed 3 = New session password does not comply with policy 9= Invalid Sendersubid or sendercompid 100 = Invalid BodyLength, session suspended

			101 = HeartBt interval too low
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2
20002	DaysToPwdExpiry		Number of days to password expiration.
	Standard Trailer	Y	

3.14.5 Logout (in/out)

The Logout message is used to gracefully disconnect a FIX session. When receiving a Logout, the counterparty should respond with a Logout. A Logout can also be the response to an unsuccessful Logon attempt.

SessionStatus = 100 means that a critical formatting error has been detected in an inbound transaction. The gateway is unable to reliably continue parsing further messages on the session. The connection is closed and can only be enabled by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 5
1409	SessionStatus		Status of the FIX session. Only set on outbound Logouts. Valid values: 3 = New session password does not comply with policy 4 = Session logout complete 5 = Invalid password or user name 6 = Account locked 8 = Password expired 9= Invalid Sendersubid or sendercompid 100 = Invalid BodyLength, session suspended 101 = HeartBt interval too low
58	Text		Free text
	Standard Trailer	Y	

3.14.6 Sequence Reset (in/out)

This message has two uses. The common usage is with GapFillFlag set to 'Y', which is used in a response to a Resend Request to indicate that a range of messages will not be resent. This is commonly used to avoid resending administrative messages like Heartbeats.

The other (very rare) usage is to reset the sequence numbers to a higher number to get out of a deadlock. This is only triggered by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 4
123	GapFillFlag		
36	NewSeqNo	Y	
	Standard Trailer	Y	

3.14.7 Resend Request (in/out)

Resend Request is used to recover messages when a sequence number gap has been detected.

Tag num	FIX Field name	Req'd	Comment
	Standard	Y	MsgType = 2

	Header		
7	BeginSeqNo	Y	
16	EndSeqNo	Y	
	Standard Trailer	Y	

3.14.8 Reject (out)

The Reject, or session-level reject, message is sent whenever the FIX gateway is able to at least partially parse the message, but the message does not adhere to the specification and cannot be delivered to the back-end system.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 3
45	RefSeqNum	Y	
371	RefTagID		
372	RefMsgType		
373	SessionRejectReason	Q	Valid values: 0 = Invalid Tag Number 1 = Required Tag Missing 2 = Tag Not Defined For This Message Type 3 = Undefined Tag 4 = Tag Specified Without A Value 5= Value Is Incorrect Out Of Range For This Tag 6 = Incorrect Data Format For Value 9 = CompID Problem 10 = SendingTime Accuracy Problem 11 = Invalid MsgType 15 = Repeating group fields out of order 16 = Incorrect NumInGroup count for repeating group 99 = Other
58	Text		
	Standard Trailer	Y	

3.14.9 Heartbeat (in/out)

A heartbeat message is sent at the interval set at Logon. It is also the response to a Test Request message.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 0
112	TestReqID		Identifier included in Test Request message to be returned in resulting Heartbeat. Required when the heartbeat is the result of a Test Request message.
	Standard Trailer	Y	

3.14.10 Test Request (in/out)

Test Request is used to "ping" the counterparty whenever a heartbeat has not arrived at the negotiated heartbeat interval.

Tag num	FIX Field name	Req'd	Comment

	Standard Header	Y	MsgType = 1
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat
	Standard Trailer	Y	

4 User Authentication

Each incoming business transaction must have a username set in the SenderSubID field. The user needs must be authenticated for the transaction to be accepted.

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

NOTE: The exchange can assign one or more FIX users of a firm as a master user.
NOTE 2: Status of suspended/resumed user's orders are subject to related business rules and definitions.

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

4.1 User Request

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

4.2 User Response

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

4.3 User Notification (Currently not supported)

The User Notification message is an unsolicited message sent when the back-end logs out a user.

4.4 Password Management (Currently not supported)

4.4.1 Renewal of passwords(Currently not supported)

A new password may be set by setting the NewPassword (925) field along with the current password in Password in the User Request message. The UserStatus (926) field of the User Response returned to the client can be checked to see if the new password was accepted.

4.4.2 Expired passwords(Currently not supported)

If the password has expired when a client tries to log in, the system will respond with a User Response message with UserStatus set to 101 – Password expired. To gain access, the client must issue a new User Request message with NewPassword set (along with the expired password in Password).
If the new password is not valid, the system will respond with another User Response with UserStatus set to 102 – New session password does not comply with policy. The client will be able to log in again with another new password.

4.5 Users across multiple sessions

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

4.6 Message Details

4.6.1 User Request (in)

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

4.6.2 User Response (out)

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

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

	Standard Trailer	Y	

4.6.3 User Notification (out)

The User Notification message is an unsolicited user status message.

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

5 Business Level Party Identifiers

5.1 Overview

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

5.1.1 Parties block

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

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

5.1.1.1 Party Identifier

The PartyID field can contain different types of identifiers. When it contains a member/participant (firm) identifier, the format is as follows:
The party identifier always consists of the two-character market code followed by the up to 5 charcater firm identifier. Example: A participant with a firm ID of XYZ, would have the party identifier of “BIXYZ”.
Available market codes:

- BI

NOTE: This party identifier scheme is also used for SenderCompID.
NOTE2: The Custodian role does not follow this format, Custodian does not contain any market code.

5.1.2 Root Parties block

In some messages a repeating group called *Root Parties* is used instead of Parties. The contents are exactly the same as for the Parties block, but the tags have new numbers, and the names of the tags are all prefixed with “Root”. The reason for this is that in some FIX messages the Parties block is in use in a repeating group. In such cases the Root Parties block is attached to the root level of the message and used instead. Currently, the Root Parties block is in use in the following messages:

- Trade Capture Report

5.2 Implicit parties

All inbound business messages must contain:

- SenderCompID (49) = party identifier of the firm entering the transaction (see section 5.1.1.1).

- SenderSubID (50) = set to a valid already authenticated username (see chapter 4 for details).

These fields implicitly identify the firm and individual entering the business message. So for all non-on-behalf-of messages, the Parties block can be omitted from the message.
NOTE: For all on-behalf-of transactions, the entering party is set in the implicit parties (SenderCompID and SenderSubID) and the executing party identifiers is set in the Parties or Root Parties block.

5.3 Available Party Roles

The following roles are used:

	Business Role	PartyRole (452)	Comment
Transaction owner = party legally responsible for consequences of the message	Firm	1 = Executing Firm	Implicit for all transactions other than on-behalf-of or trade reports. Reporting party in trade reports.
	Individual user	12 = Executing Trader	Implicit for all transactions other than on-behalf-of.
Transaction Sender = initiator party of the transaction message	Firm	7 = Entering Firm	Firm who has recorded or reported the execution
	Individual user	36 = Entering Trader	Trader who initiates/submits the transaction (i.e. associated with Entering Firm)
Counterparty in Trade Capture Reports	Firm	17 = Contra Firm	Counterparty in Trade Capture Reports.
Custodian	AFK	28 = Custodian	Optional AFK identifier. Refer to Account Model functionality.
		14 = Giveup Clearing Firm (Takeup Firm)	
		38 = Position Account	
		83 = Clearing Account	
		1001 = Confirmed by Firm	
		1002 = Confirmed by User	
		1003 = Reported by Firm	
		1004 = Reported by User	
		1005 = Affirmed by Firm	
		1006 = Affirmed by User	
		1007 = Give-up Account	

NOTE: individual users are not used to identify reporting party or counterparty in Trade Capture Reports.

5.4 On Behalf of Identifiers

- All inbound business messages sent on behalf of another party must include the Parties block. Two parties must be present in each on behalf of transaction: PartyRole = 1, Executing Firm set to the id of the firm the transaction is entered on behalf of (legal owner).
- PartyRole = 12, Executing Trader set to the id of the trader the transaction is entered on behalf of (legal owner).
  The party entering the transaction is set in the implicit party identifier fields (SenderCompID and SenderSubID).

6 Order Management

6.1 Overnight orders

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

6.2 Pass-thru fields

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

Field	Tag	Mapped to	Length	Comment
Account	1	ex_client or exchangeInfo	10/15	Client Account. Refer to Account model configured for the order book.
Custodian Party	448	exClinet or n/a	3	AFK Account. Refer to Account model configured for the order book.
AllocID	70	customer_info	15	

NOTE: The pass-thru fields are *not* echoed back on rejects.

6.3 Clearing Accounts

Clearing Account is generally split into two parts:

- Clearing Firm
- Clearing Account

The fields used for Clearing Account information is a bit different in different scenarios. Clearing Account information is carried in different fields depending on:

- The direction of the message (inbound to the exchange or outbound)
- The type of message

The following matrix shows which fields should be used in which scenario:

Messages	Direction	Clearing Account	Clearing Firm	Comment
New Order Single Cancel Replace Request Execution Report	IN/OUT	Account (1)	implied	Clearing Firm is the same as the owner of the order.
Trade Reports	IN/OUT	Account (1)	implied	Clearing Firm is the same as the owner of

				the trade.
Rectify Trade	IN	AllocAccount (79)	Nested2PartyID *(Nested2PartyRole=4)*	
Give-ups (Allocation Instruction, Allocation Report)	IN/OUT	AllocAccount (79)	NestedPartyID *(NestedPartyRole=14)*	

6.4 Instrument Identifiers

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

- Symbol (55) which should contain the OMNet short name (ins_id_s) for the security.
- SecurityID (48) containing the Orderbook ID of the security. This is an alternative numeric identifier that can be used instead of Symbol. **NOTE:**
    - The Orderbook ID identifier is **not** provided via OMNet Reference Data.
    - The Orderbook ID *can* be different across trading days for the same security.

6.5 Main Workflow

6.5.1 New Order

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

6.5.2 Fills

When an order is filled the Execution Report will contain details about the fill. See section 6.10.15 for message details. In addition, a Trade Capture Report will be produced. The principal differences between the two are:

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

Trade Capture Reports are messages capturing the trade as such and is primarily intended for downstream processing. The Trade Capture Report is used to inform a variety of parties about a trade, e.g.: broker back office; clearing firms; clearing houses; depositories and; regulators. As such downstream processing occurs at various locations and for different purposes, the Trade Capture Report message might look slightly different depending on the receiver.
Trade Capture Report messages are also used for a large number of other purposes, including reporting of privately negotiated trades and relaying trades to parties not directly involved in the trade – but this is outside the scope of this chapter.

Trade reversals and corrections are only sent as Trade Capture Reports.

6.5.2.1 Trade Match ID

The TrdMatchID (880) contains the match id generated by the system. TrdMatchID will hold the 64 bit binary match id encoded as a 16 byte hex string.**NOTE:** TrdMatchID is also set in Trade Capture Report confirmation messages.

6.5.3 Order Modification

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

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

6.5.3.1 Order Attributes allowed to change

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

- OrderQty (38)
- MaxFloor (111)
- TimeInForce (59) together with ExpireDate (432) or TradingSessionID (336)
- AllocID (70), pass-thru field
- Price (44)

NOTE: Any change to the price of an order, or increasing quantities will result in the order losing its priority in the market.
NOTE 2: Modifying an order to TimeInForce = IOC or FoK is not allowed.
NOTE 3: Modifying the price of an order to a zero is not allowed. If a zero price is desired, the order has to be deleted and a new order with price 0 entered.
NOTE 4: If MaxFloor, PartyID (where PartyRole is Custodian) or TimeInForce are not intended to be changed, **do not include them** in the Cancel Replace message. They may cause the order to loose priority or the Cancel Replace to be rejected.

6.5.3.2 Restatements

The Execution Report – Restatement message is used for restating the overnight orders (GTC/GTD) in the morning, In this case, the ExecRestatementReason will be set to 1 = GT renewal / restatement (no corporate action). See section 6.10.14 for message details.

6.5.3.3 Unsolicited modification of orders entered via FIX

Orders entered via FIX can be modified via other protocols like Omnet. It may also be possible for the marketplace to modify existing orders. In such an event an Execution Report – Unsolicited Order Update will be sent out over FIX. See section 6.10.13 for message details.

6.5.4 Order Cancellation

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

- Execution Reports are issued relaying the status of every canceled order.
- In some cases orders may be cancelled in the system without prior request by the user. These will be sent as an Execution Report – Unsolicited Cancel to the client.
- The system will generate cancel messages (Execution Report –IOC/Fok Order Cancel) for every IOC and FoK order.
- The system will generate cancel messages (Execution Report – Market-to-Limit Order Cancel) for Market-to-Limit orders that could not be immediately matched.
- The Order Cancel Request cannot be used for partial cancels.

6.5.4.2 Cancellation of orders **not** sent in via FIX

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

6.5.4.3 Unsolicited cancellation of orders entered via FIX

Orders entered via FIX may be cancelled via other protocols like Omnet, or possibly by the marketplace. In such an event an Execution Report – Unsolicited Cancel will be sent out over FIX. See section 6.10.12 for message details.

6.5.5 Order suspension/inactivation at connection loss

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

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

Upon reconnection, Execution Reports will be sent out for all suspended orders. The Execution Reports will have OrdStatus set to 9 – Suspended. See section 6.10.16 for message details.
Suspended orders may be cancelled using ordinary Order Cancel Request messages. Suspended order cannot be activated again.
NOTE: The Execution Report –Order Suspended will not contain TargetSubID (57).
NOTE 2: Suspended orders will be cancelled at end-of-day.

6.5.6 Paused Orders

Orders that are paused by Matching Engine and are not eligible to matching. These orders do not contribute to the EP price and is not part of the price picture in the market. ME also automatically reinstates such paused orders with their original time priorities when the conditions allow it.

Matching Engine acknowledges orders as paused when :

- Orders that move outside the effective price limit range
- Non-off-hours orders in off-hours sessions

6.5.6.1 Pausing orders that move outside price limits

In this example there are two orders, #1 and #2 at different prices.

a) Initially the price limits are not enabled and both orders are active.
b) Price limits are enabled. Both orders are inside the price limits and stay active.

c) The reference price is changed and the upper/lower price limits move. Order #1 is still inside the limits and remain active. Order #2 falls outside the price limits and is automatically paused.
d) Price limits are disabled. Order #2 is reinstated with its original time priority.
e) Price limits are enabled again. Order #2 falls outside the lower limit and is automatically paused.

6.5.6.2 Pausing orders outside price limits and in the off-hours session

Note that this example may not be a realistic business scenario. It is included here for the purpose of explaining the order pausing mechanism.

In this example there are three orders at different prices. Order #1 is an off-hours order and orders #2 and #3 are non-off-hours orders.

a) Initially the price limit monitoring is not enabled and all orders are active.
b) Price limits are enabled. All orders are inside the price limits and stay active.
c) The reference price is changed and the upper/lower price limits move. Orders #1 and #2 are still inside the limits and remain active. Order #3 falls outside the price limits and is automatically paused.
d) The instrument goes into off-hours mode. Orders #2 and #3 are non-off-hours orders and shall not participate. Order #2 is paused automatically. Order #3 is already in paused state and doesn’t need to be changed.
e) The instrument remains in off-hours mode. The price limits are disabled, however since both O#2 and O#3 are non-off-hours orders they shall stay paused.
f) The instrument leaves the off-hours mode, and is now trading without price limit supervision in the normal session. The non-off-hours orders O#2 and O#3 are automatically reinstated.
g) Price limits are enabled, and since O#3 is outside the limits it gets automatically paused again.

When an outlier[1] order is inserted and is immediately paused, or order is paused by system due to session change, fix gateway will publish an ER identified by OrdStatus(39) = Suspended(9) and Text(58) = "Paused", indicating that order is paused by system.

6.6 Order Features

6.6.1 Order Identification

6.6.1.1 Client Order ID

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

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

[1] An outlier order is either an offer priced above the upper price limit or a bid priced below the lower price limit.

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

Client Order IDs when the Firm uses multiple FIX sessions

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

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

6.6.1.2 Order ID

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

NOTE: Genium INET OrderIDs are only unique *per orderbook and Side*. So a buy and a sell order in the same orderbook may have the same OrderID (As it is in the case of Mass Quotes). Care must be taken to base identification of orders on OrderID, orderbook id (SecurityID/Symbol), and Side.

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

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

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

6.6.1.3 Execution ID

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

6.6.1.4 ExecType

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

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

6.6.2 Order States

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

An order can be in the following intermediate states:

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

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

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

6.6.3 Order Types

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

- Market
- Limit
- Market-to-Limit (called Market with leftover as limit in FIX).

6.6.3.1 Market Orders

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

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

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

6.6.3.2 Market-to-Limit Orders

A Market-to-Limit order is a market order where the remaining quantity is placed in the book at the price which part of the order was executed. If there is no order on the opposing side, the Market-to-Limit order will be cancelled immediately.

In comparison to a Market order, the Market-to-Limit order only executes at the best price level and therefore does not trade through the book.

By setting TimeInForce to IOC or FoK, the Market-to-Limit order will behave like a Market order but only match at the highest price level.

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

NOTE2: Market to Limit Orders entered during an auction state that does not match will have all of its quantity converted to a limit order after the uncross. In this scenario an Unsolicited update message will be sent out.

6.6.4 Order Expiry

An order can specify various conditions for when or how it should expire or be automatically removed from the book.
The evening when a GTD order expires, an Execution Report with OrdStatus (39) set to Cancelled will be sent out for that order. See section 6.10.17 for message details.
A GTC order can also expire. Example: A GTC order is suspended. If it isn't deleted or reactivated (not possible via FIX) the same day, an ER with OrdStatus set to Cancelled will be sent out the next day.
NOTE: Only if a GTC order expires because the instrument expires intra-day, or if a GTD order is cancelled by a corporate action, an order expired transaction will be sent out.
Supported TimeInForce (59) values:

Value	Name	Comment
0	Day	
1	Good Till Cancel (GTC)	
3	Immediate Or Cancel (IOC)	
4	Fill or Kill (FoK)	FoK orders cannot have MatchIncrement (1089) set.
6	Good Till Date (GTD)	GTD orders must have ExpireDate (432) set.
S	Good till End of Session (GTS)	Order expires at the end of the *first* session with the state type set in TradingSessionID (336). See detailed description below.
9	At Crossing	Used for entering Imbalance orders. Tag 44 (Price) must be set to 0 and Tag 40 (OrdType) must be set to 1 (Market).

6.6.5 Good till End of Session (GTS)

GTS is a Extension to FIX. A GTS order is similar to a GTD order, but instead of setting the date of expiry, the session type when the order shall expire is set. TimeInForce nust be set to S.

A GTS order is valid until the end of the *first* session of the type given in TradingSessionID (336). For valid session values, contact the exchange.
The session type is used to group several sessions with equal order behavior into one entity to be used for session related order handling. In the cases where the same session type is run multiple times during a day, the order will expire on the earliest possible occasion.

6.6.6 Quantity Conditions

An order can specify various types of quantity conditions.
Match Increment (1089). In this solution, MatchIncrement is used to enter Block or Odd Lot orders. By setting MatchIncrement to the number of units configured as the Block Lot/Odd Lot for the orderbook of the order (Lot size), a block lot/odd lot order can be entered. The Lot sizes for a particular instrument must be looked up in Reference Data (this is **not** provided via FIX Order Entry).
An order with an incorrect MatchIncrement value will be rejected.
If MatchIncrement is not set, the order defaults to Round Lot.
NOTE: MatchIncrement is not allowed for FoK orders.

All-or-None (AON) is an instruction to fill an order completely or not at all; similar to a FoK but it remains in the book if not executed immediately. Tag 18, ExecInst is used for AON orders.
NOTE: AON orders cannot have MatchIncrement set.

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

MaxFloor (111): Used to indicate the maximum order quantity shown in the public Market Data.
NOTE: MaxFloor = 0, a completely hidden order, is **not** supported in Genium INET. Setting MaxFloor to zero will make the full order visible.
NOTE 2: MaxFloor has been changed to behave as expected in standard FIX. The expected behavior is for MaxFloor value to be decreased when the order is partially traded. The previous implementation kept MaxFloor on the original value.

6.6.7 Triggering Instructions

The Triggering Instructions block in FIX is used to express predefined automatic order modifications. Triggers can act on different events. The TriggerType (1100) field determines what should trigger a change. The only action supported is for the triggered order to be *activated*. The trigger order remains hidden and inactive until the trigger condition is met. When the trigger hits, the order is either traded or inserted into the book as if it was a new order.
Only one triggering instruction is allowed per order. All the order attributes available for a "normal" order (e.g. Order Type, Time In Force etc) are supported for the order to be triggered.

"NOTE: Trigger orders, which have expiration day=current day, will be removed at the end of day if the triggering condition has not yet been met." The following fields can be used:

Tag no	Name	Comment
1100	TriggerType	Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggerAction	Defines the type of action to take when the trigger hits. Valid values: 1 = Activate
1102	TriggerPrice	A specified limit price to validate against price movements –the trigger hits when the price is reached.
1113	TriggerTradingSessionID	Identifier of the session type when the order is to be triggered. StateIDs are sent out on fix reference data in tag 20032.
1103	TriggerSymbol	Symbol used for price triggers
1104	TriggerSecurityID	Identifier of the security used for price triggers.
1105	TriggerSecurityIDSource	Valid values: M = Marketplace-assigned identifier
1107	TriggerPriceType	Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggerPriceDirection	Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices. 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.

By combining TriggerPriceType and TriggerPriceDirection the following Trigger Instructions can be achieved.

TrigPriceType	TrigPriceDirection	Meaning
Best Offer	U	Ask >=
Last Trade	U	Last >=
Best Bid	U	Bid >=
Best Offer	D	Ask <=
Last Trade	D	Last <=
Best Bid	D	Bid <=

6.6.7.1 Price Triggering

When an order is matched, stored, altered, expired or deleted affecting the Best Bid Offer (BBO) or the Last Match Price of the matching engine, the system checks for any "non-triggered" orders having a condition that is now met. It is possible to trigger off price movements occurring in the same orderbook or in a different orderbook.

The following fields must be set for a price trigger:

- TriggerType (1100) set to *Price Movement*
- TriggerAction (1101) set to *Activate*
- TriggerPrice (1102) set to the triggering price
- TriggerSymbol (1103) OR TriggerSecurityID+TriggerSecurityIDSource
- TriggerSecurityID (1104) set to the triggering instrument
- TriggerSecurityIDSource (1105)
- TriggerPriceType (1107) to specify the price type; *best bid, best offer or last trade.*
- TriggerPriceDirection (1109) to indicate price movement direction

6.6.7.2 Triggering off Session Changes

Triggering off Session Changes can be used to activate an order at a specified session. The following fields need to be set:

- TriggerType (1100) set to *Specified Trading Session*
- TriggerAction (1101) set to *Activate*
- TriggerTradingSessionID (1113) Identifier of the trading session to activate order at. StateTypeID are sent on ref data in tag 20032.

6.6.7.3 Triggering workflow

A trigger order can go into three different states at entry:

- Not activated – the order is not immediately triggered, and is placed outside of the book waiting to be triggered.
- Immediately activated, immediately filled
- Immediately activated, placed on book – the order is immedialtely triggered but does not immediately trade.

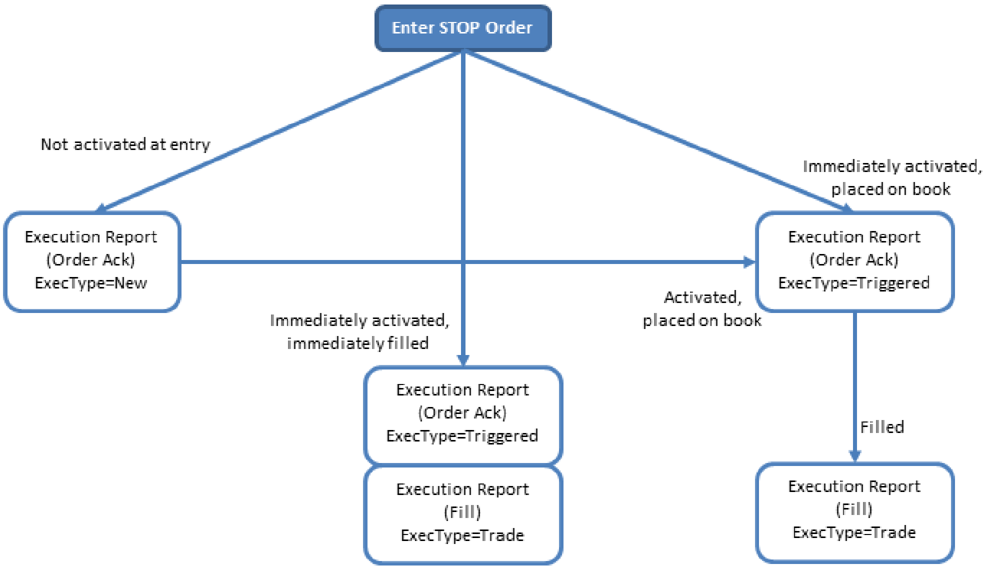

Enter STOP Order
Not activated at entry
Immediately activated, placed on book
Execution Report (Order Ack) ExecType=New
Execution Report (Order Ack) ExecType=Triggered
Activated, placed on book
Immediately activated, immediately filled
Execution Report (Order Ack) ExecType=Triggered
Filled
Execution Report (Fill) ExecType=Trade
Execution Report (Fill) ExecType=Trade

Figure 1, Trigger Order states and resulting messages

To understand the state of a trigger order, the client needs to examine the ExecType (150) field of the Execution Report messages received.

- ExecType=New (0) means the order was not activated on entry.
- ExecType=Triggered (L) means the order was activated at entry.

ExecType=Trade (F) means the order was partially or fully traded. A trigger order will always be activated before it trades.

6.6.7.4 Cancellation of Trigger Orders

An order with a trigger condition can be cancelled using the ordinary Order Cancel Request message. As a consequence of the back-end having different cancel messages for triggered and not yet triggered orders, it is unlikely but possible for a cancel of a non-triggered order to be rejected while the order is left in the book. This can only happen *once*, if the order is triggered while the cancel is sent in. *In this (unlikely) event, a second cancel must be sent for the same order.*

6.6.8 Short Sell Orders

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

6.6.9 Imbalance Orders

The Trading system supports entering of Imbalance Orders. To enter an Imbalance order the following information must be sent in a New Order Single:
Tag 59 (TimeInForce) = 9 (At crossing)
Tag 44 (Price) = 0
Tag 40 (OrdType) = 1 (Market)

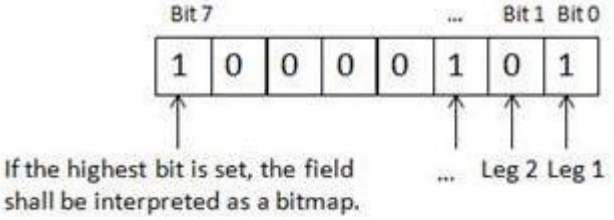
6.6.10 Open Close Indicator

The OpenCloseIndicator(20199) is a extension to standard FIX. The open/close indicator in can be set on inbound orders, cancel replaces or reported trades to indicate if the resulting position after a trade should be an opening or a closing position. The following values are supported:
0 = Default for the account (default value) / No change (for replace transactions)
1 = Open
2 = Close
4 = Set to default for account (for replace transactions)

For combination instruments, the OpenCloseIndicator can be used to control the open/close effect *per leg*. This is not possible with the standard FIX PositionEffect (77) field, which is why this extension had to be added. Values from 128 and above are used for per leg open/close indicators.
For per leg functionality, it is easiest to view the field as a bit vector (8 bits). The highest bit (bit 7 – decimal 128) is used to indicate that the per leg functionality is used.

Each leg that makes up the combination has one bit to specify if the resulting position should be opening or closing. Bit 0 is used for the first leg, bit 1 for the second leg etc.

- A bit set to 0 (zero) indicates open.
- A bit set to 1 indicates close.


Bit 7
...
Bit 1
Bit 0
1
0
0
0
0
1
0
1
If the highest bit is set, the field shall be interpreted as a bitmap.
...
Leg 2
Leg 1

Example:

The above example, where Legs 1 and 3 were “close”, and legs 2 and 4 were “open” (binary 10000101) would be represented in FIX as:

20199=133

The values entered in the OpenCloseIndicator field will be returned in Execution Reports. Trade confirmations will contain the standard PsoitionEffect field. That’s sufficient as trade confirmations are generated per leg for a combination trade.

6.6.11 On Close Orders

To enter on close order, you enter a session state order (SSO) with time validity IOC (FAK) and order type market. SSO orders are entered by using Trigger on Session functionality.
Please see Triggering Instruction section for more details.

To enter an on close order the following information must be sent in a New Order Single:
Tag 59 (TimeInForce) = 3 (IOC)
Tag 44 (Price) = 0
Tag 40 (OrdType) = 1 (Market)

- TriggerType (1100) set to *Specified Trading Session*
- TriggerAction (1101) set to *Activate*
- TriggerTradingSessionID (1113) Identifier of the trading session to activate order at. StateTypeID are sent on fix reference data in tag 20032.

NOTE: On close orders should be sent without off-hour attribute in order to execute properly.

6.7 Missing required fields in Rejects

Due to the way the back-end works, certain fields required in standard FIX 5.0 SP2 for application-level rejects will be missing.
For Order rejects (Execution Report – reject), the following required field will not be present:
Side (54)

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

6.8 Business Message Reject

The Business Message Reject is used to report rejections in situations where other reject messages are not available, e.g. when the inbound message does not reach the trading engine due to trading being closed or authorization not sufficient. See section 6.10.18 for message details.
NOTE: The user must be prepared to receive this message as an alternative response to all other business messages.
NOTE: If the message count reach pacing limit then all the following messages will be rejected with the Business Message Reject.

6.9 How to interpret the message details listings

6.9.1 How to interpret the Required (Req'd) column

A 'Y' marks the field as required in standard FIX (and of course also in this implementation). A 'Q' means that the field is required in this implementation although it is not required in standard FIX. No entry at all means the field is optional.

6.9.2 Repeating groups

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

453	NoPartyIDs				Optional repeating group only used for on behalf of transactions and account model values.
→	448	PartyID		Q	Party identifier.
→	447	PartyIDSource		Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified.
→	802	NoPartySubIDs			Number of PartySubIDs present. Only used for PartyRole=Executing Firm. Will always be 1.
→	→	523	PartySubID	Q	Sub-identifier of party.
→	→	803	PartySubIDType	Q	Type of PartySubID (523) value

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

6.10 Message Details

6.10.1 New Order Single –inbound to Marketplace (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = D
11	ClOrdID		Y	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 12 = Executing Trader 14 = Giveup Clearing Firm 28 = Custodian

			83 = Clearing Account
1	Account	Q	Required pass-thru field set by client and echoed back by marketplace.

528	OrderCapacity	Q	Account type
			Valid values:
			A = Customer
			P = Principal
			F = Fund Order
			C = Primary Dealer Corp
			K = Primary Dealer Govt
			O = Portfolio Mgmt Company
			T = Investment Trust
			Y = Other Fund
			S = Special Client
			N = Other 2
			D = Other 3
			NOTE: OrderCapacity is not used by derivatives market

111	MaxFloor		For hidden orders.
21	HandlInst		Instructions for order handling on Broker trading floor. Valid values: 1 = Automated execution order, private, no Broker intervention (default value)
18	ExecInst		Valid values: G = All or None (AON)
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime	Y	
38	OrderQtyData/OrderQty	Y	
40	OrdType	Y	Valid values: 1 = Market 2 = Limit K = Market With Left Over as Limit (market order with unexecuted quantity becoming limit order at last price)
44	Price		Required for Limit orders
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
1100	TriggeringInstruction/ TriggerType		Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggeringInstruction/ TriggerAction		Defines the type of action to take when the trigger hits. Valid values: 1 = Activate

1102	TriggeringInstruction/ TriggerPrice		A specified limit price to validate against price movements –the trigger hits when the price is

				reached.
1113	TriggeringInstruction/ TriggerTradingSessionID			Identifier of Trading Session when the order is to be triggered.
1103	TriggeringInstruction/ TriggerSymbol			Symbol used for price triggers
1104	TriggeringInstruction/ TriggerSecurityID			Identifier of the security used for price triggers.
1105	TriggeringInstruction/ TriggerSecurityIDSource			SecurityIDSource of the instrument used for price triggering. Valid values: M = Marketplace-assigned identifier
1107	TriggeringInstruction/ TriggerPriceType			Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggeringInstruction/ TriggerPriceDirection			Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices. 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.
59	TimeInForce		Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
386	NoTradingSessions			Only set for GTS and Off-hour orders. Can only be set to 1
→	336	TradingSessionID	Q	State type of order expiration. Conditionally required if TimeInForce = GTS. Valid Values: StateTypeId for GTS or ‘A’ for Off-hour Orders.
432	ExpireDate			Date of order expiration. Conditionally required if TimeInForce = GTD
70	AllocID			Optional pass-thru field set by client and echoed back by marketplace.
1089	MatchIncrement			If set, must be set to correct block lot/odd lot size, otherwise it will be rejected.
1094	PegPriceType			Valid values: 2 = Mid-price peg
20199	OpenCloseIndicator			The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open

			2 = Close Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.2 Order Cancel Request (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = F
41	OrigClOrdID		Y	Set to "NONE" if using OrderID instead.
37	OrderID			Recommended to be used instead of OrigClOrdID.
11	ClOrdID		Y	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group only used for on behalf of transactions.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 12 = Executing Trader
55	Instrument/Symbol			OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
48	Instrument/SecurityID			Orderbook ID
22	Instrument/SecurityIDSource			Valid values: M = Marketplace-assigned identifier
54	Side		Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime		Y	
38	OrderQtyData/OrderQty		Y	**NOTE:** Required in FIX but ignored by the system. Partial cancels are not supported.
	Standard Trailer		Y	

6.10.3 Order Cancel Replace Request (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = G
37	OrderID			Recommended to be used instead of OrigClOrdID.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID

			specified. Valid values: 1 = Executing Firm 12 = Executing Trader 14 = Giveup Clearing Firm28 = Custodian 83 = Clearing Account
41	OrigClOrdID	Y	ClOrdID of the order to modify/cancel. Set to “NONE” if using OrderID instead.
11	ClOrdID	Y	Unique identifier set by the client.
1	Account		Required pass-thru field set by client and echoed back by marketplace.

528	OrderCapacity		Account type
			Valid values:
			A = Customer
			P = Principal
			F = Fund Order
			C = Primary Dealer Corp
			K = Primary Dealer Govt
			O = Portfolio Mgmt Company
			T = Investment Trust
			Y = Other Fund
			S = Special Client
			N = Other 2
			D = Other 3
			NOTE: OrderCapacity is not used by derivatives market

70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
111	MaxFloor		For hidden orders.
21	HandlInst		Instructions for order handling on Broker trading floor. Valid values: 1 = Automated execution order, private, no Broker intervention (default value)
18	ExecInst		Valid values: G = All or None (AON)
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
54	Side	Y	Required in FIX, but not allowed to change Valid values: 1 = Buy 2 = Sell 5 = Sell Short
60	TransactTime	Y	
38	OrderQtyData/OrderQty	Y	
40	OrdType	Y	Required in FIX, but not allowed to change Valid values: 1 = Market 2 = Limit
44	Price		Required for Limit orders
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.

59	TimeInForce		Valid values:

			0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = No change 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.4 Order Cancel Reject (out)

Purpose: Reject of Order Cancel Request.
Identified by: MsgType = 9 AND CxlRejResponseTo = 1

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

6.10.5 Order Cancel Reject – Cancel Replace (out)

Purpose: Reject of Order Cancel Replace Request.
Identified by: MsgType = 9 AND CxlRejResponseTo = 2

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

6.10.6 Execution Report – Order Ack (out)

Purpose: Order Acknowledgement.
Identified by: MsgType = 8 AND ExecType = (0 or L)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	
11	ClOrdID		Q	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader

			14 = Giveup Clearing Frim 28 = Custodian 36 = Entering Trader 83 = Clearing Account
17	ExecID	Y	
150	ExecType	Y	Valid values: 0 = New L = Triggered or Activated by the system
39	OrdStatus	Y	Valid values: 0 = New
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit K = Market With Left Over as Limit (market order with unexecuted quantity becoming limit order at last price)
44	Price		
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
1100	TriggeringInstruction/ TriggerType		Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggeringInstruction/ TriggerAction		Defines the type of action to take when the trigger hits. Valid values: 1 = Activate
1102	TriggeringInstruction/ TriggerPrice		A specified limit price to validate against price movements –the trigger hits when the price is reached.
1113	TriggeringInstruction/ TriggerTradingSessionID		Identifier of Trading Session when the order is to be triggered.
1103	TriggeringInstruction/ TriggerSymbol		Symbol used for price triggers
1104	TriggeringInstruction/ TriggerSecurityID		Identifier of the security used for price triggers.
1105	TriggeringInstruction/ TriggerSecurityIDSource		SecurityIDSource of the instrument used for price triggering. Valid values: M = Marketplace-assigned identifier

1107	TriggeringInstruction/ TriggerPriceType		Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggeringInstruction/ TriggerPriceDirection		Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices. 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.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 3 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst		Valid values: G = All or None (AON)
151	LeavesQty	Y	Will be equal to OrderQty on Order.
14	CumQty	Y	Will be 0 on Order Ack.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
1089	MatchIncrement		Block Size
1094	PegPriceType		Valid values: 2 = Mid-price peg
111	MaxFloor		For hidden orders. Contains currently shown quantity.
119	SettlCurrAmt		Order value.
20015	OptionPremiumAmt		Premium value
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10

			for details.
	Standard Trailer	Y	

6.10.7 Execution Report – IOC/FoK Order Cancel (out)

Purpose: Cancel of IOC or FOK order. Will always be sent last in a sequence following any immediate fills.

Identified by: MsgType = 8 AND ExecType = 4 AND TimeInForce = 3 OR 4

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	
11	ClOrdID	Q	Unique identifier set by the client.
453	NoPartyIDs		Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448 PartyID	Q	Party identifier.
→	447 PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452 PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 3 = Entering Firm 12 = Executing Trader 28 = Custodian 36 = Entering Trader
17	ExecID	Y	
150	ExecType	Y	Valid values: 4 = Canceled
39	OrdStatus	Y	Valid values: 4 = Canceled
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit
44	Price		
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
59	TimeInForce	Q	Valid values: 3 = Immediate Or Cancel (IOC)

			4 = Fill Or Kill (FoK)
151	LeavesQty	Y	Will be 0.
14	CumQty	Y	
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders.
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
	Standard Trailer	Y	

6.10.8 Execution Report – Market to Limit Order Cancel (out)

Purpose: Sent if a Market to Limit order cannot be immediately executed (nothing on opposite side of the order book).

Identified by: MsgType = 8 AND ExecType = 4 AND OrdType = K

Tag	FIX tag name		Req’d	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	
11	ClOrdID		Q	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier..
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 3 = Entering Firm 12 = Executing Trader 28 = Custodian 36 = Entering Trader
17	ExecID		Y	
150	ExecType		Y	Valid values: 4 = Canceled
39	OrdStatus		Y	Valid values: 4 = Canceled
1	Account			Optional pass-thru field set by client and

			echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: K = Market With Left Over as Limit (market order with unexecuted quantity becoming limit order at last price)
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 3 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
151	LeavesQty	Y	Will be 0.
14	CumQty	Y	Will be 0 in this case.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders.
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.9 Execution Report – Order Reject (out)

Purpose: Order reject.

Identified by: MsgType = 8 AND ExecType = 8

NOTE: This message lacks the required Side (54) field.
NOTE 2: The Symbol field is set to [N/A].

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	"NONE"
11	ClOrdID	Q	Unique identifier set by the client.
17	ExecID	Y	
150	ExecType	Y	Valid values: 8 = Rejected
39	OrdStatus	Y	Valid values: 8 = Rejected
103	OrdRejReason	Q	Valid values: 0 = Broker / Exchange option
55	Instrument/Symbol	Q	Will be set to [N/A]
151	LeavesQty	Y	Will be 0 on Order Reject.
14	CumQty	Y	Will be 0 on Order Reject.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
58	Text		Error message
	Standard Trailer	Y	

6.10.10 Execution Report – Cancel Replace Ack (out)

Purpose: Acknowledgement of Order Cancel Replace Request.
Identified by: MsgType = 8 AND ExecType = 5

Tag	FIX tag name		Req’d	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	Genium INET order_number
11	ClOrdID		Q	Unique identifier set by the client.
41	OrigClOrdID			ClOrdID of the order to modify/cancel.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 3 = Entering Firm 12 = Executing Trader 14 = Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account
17	ExecID		Y	
150	ExecType		Y	Valid values:

			5 = Replaced
39	OrdStatus	Y	Valid values: 0 = New 1 = Partially Filled 2 = Filled 4 = Canceled
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit
44	Price		
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
1100	TriggeringInstruction/ TriggerType		Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggeringInstruction/ TriggerAction		Defines the type of action to take when the trigger hits. Valid values: 1 = Activate
1102	TriggeringInstruction/ TriggerPrice		A specified limit price to validate against price movements –the trigger hits when the price is reached.
1113	TriggeringInstruction/ TriggerTradingSessionID		Identifier of Trading Session when the order is to be triggered.
1103	TriggeringInstruction/ TriggerSymbol		Symbol used for price triggers
1104	TriggeringInstruction/ TriggerSecurityID		Identifier of the security used for price triggers.
1105	TriggeringInstruction/ TriggerSecurityIDSource		SecurityIDSource of the instrument used for price triggering. Valid values: M = Marketplace-assigned identifier
1107	TriggeringInstruction/ TriggerPriceType		Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggeringInstruction/ TriggerPriceDirection		Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices. 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.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders.
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst		Valid values: G = All or None (AON)
151	LeavesQty	Y	
14	CumQty	Y	
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
1089	MatchIncrement		Block Size
111	MaxFloor		For hidden orders. Contains currently shown quantity.
119	SettlCurrAmt		Order value.
20015	OptionPremiumAmt		Premium value
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.11 Execution Report – Cancel Ack (out)

Purpose: Acknowledgement of Order Cancel Request.

Identified by: MsgType = 8 AND ExecType = 4

Tag	FIX tag name	Req’d	Comment

	Standard Header		Y	MsgType = 8
37	OrderID		Y	Genium INET order_number
11	ClOrdID		Q	Unique identifier set by the client.
41	OrigClOrdID			ClOrdID of the order to modify/cancel. Will not be set for orders not entered via FIX.
453	NoPartyIDs			Optional repeating group only used for on behalf of transactions.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account
17	ExecID		Y	
150	ExecType		Y	Valid values: 4 = Canceled
39	OrdStatus		Y	Valid values: 4 = Canceled
1	Account			Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity			Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol		Q	OMNet short name.
48	Instrument/SecurityID		Q	Orderbook ID
22	Instrument/SecurityIDSource		Q	Valid values: M = Marketplace-assigned identifier
54	Side		Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty		Q	
151	LeavesQty		Y	Will be 0 on Cancel Ack.
14	CumQty		Y	
6	AvgPx		Y	Always set to 0.0
60	TransactTime		Q	
70	AllocID			Optional pass-thru field set by client and echoed back by marketplace.
119	SettlCurrAmt			Order value.
20015	OptionPremiumAmt			Premium value
797	CopyMsgIndicator			Set to ‘Y’ on Drop Copy messages
20199	OpenCloseIndicator			The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value)

			1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.12 Execution Report – Unsolicited Cancel (out)

Purpose: Order was cancelled outside of FIX (via other protocol or by the marketplace). GTD, GTS **or GTC** Order Canceled. For GTD orders the Cancel transaction will be sent at the end of the daythe order expired. For GTS orders the Cancel message is sent at order expiry. Order expiry can occur for GTC orders under certain conditions. See Order Expiry section for details

Identified by: MsgType = 8 AND ExecType = 4 AND ExecRestatementReason = 8 OR ExecRestatementReason = 4

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	Genium INET order_number
11	ClOrdID	Q	Unique identifier set by the client.
453	NoPartyIDs		Optional repeating group used for on behalf of transactions.
→	448 PartyID	Q	Party identifier.
→	447 PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452 PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account
17	ExecID	Y	
150	ExecType	Y	Valid values: 4 = Canceled
39	OrdStatus	Y	Valid values: 4 = Canceled
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy

			2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
151	LeavesQty	Y	Will be 0 for a canceled order
14	CumQty	Y	
6	AvgPx	Y	Always set to 0.0
60	TransactTime	Q	
378	ExecRestatementReason	Q	Valid values: 4 = Broker option 8 = Market (Exchange) option
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
119	SettlCurrAmt		Order value.
20015	OptionPremiumAmt		Premium value
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
58	Text		Information on why the order was canceled
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.13 Execution Report – Unsolicited Order Update (out)

Purpose: Order was updated outside of FIX (via other protocol or by the marketplace).
Identified by: MsgType = 8 AND ExecType = D AND ExecRestatementReason = 8

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	
11	ClOrdID		Q	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account

17	ExecID	Y	
150	ExecType	Y	Valid values: D = Restated
39	OrdStatus	Y	Valid values: 0 = New 1 = Partially Filled
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	Order quantity
40	OrdType	Q	Valid values: 1 = Market 2 = Limit
44	Price		Order price
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
1100	TriggeringInstruction/ TriggerType		Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggeringInstruction/ TriggerAction		Defines the type of action to take when the trigger hits. Valid values: 1 = Activate
1102	TriggeringInstruction/ TriggerPrice		A specified limit price to validate against price movements –the trigger hits when the price is reached.
1113	TriggeringInstruction/ TriggerTradingSessionID		Identifier of Trading Session when the order is to be triggered.
1103	TriggeringInstruction/ TriggerSymbol		Symbol used for price triggers
1104	TriggeringInstruction/ TriggerSecurityID		Identifier of the security used for price triggers.
1105	TriggeringInstruction/ TriggerSecurityIDSource		SecurityIDSource of the instrument used for price triggering. Valid values: M = Marketplace-assigned identifier
1107	TriggeringInstruction/ TriggerPriceType		Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggeringInstruction/ TriggerPriceDirection		Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices.

			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.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders.
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6= Good Till Date (GTD)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst		Valid values: G = All or None (AON)
151	LeavesQty	Y	
14	CumQty	Y	
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
378	ExecRestatementReason	Q	Valid values: 4 = Broker option 8 = Market (Exchange) option
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
119	SettlCurrAmt		Order value.
20015	OptionPremiumAmt		Premium value
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
1089	MatchIncrement		Block Size
111	MaxFloor		For hidden orders. Contains currently shown quantity.
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.14 Execution Report – Restatement (out)

Purpose: Restatement of overnight (GTC/GTD) orders in the morning.

Identified by: MsgType = 8 AND ExecType = D AND ExecRestatementReason = 1

Tag	FIX tag name	Req’d	Comment

	Standard Header		Y	MsgType = 8
37	OrderID		Y	Genium INET order_number
11	ClOrdID		Q	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account
17	ExecID		Y	
150	ExecType		Y	Valid values: D = Restated
39	OrdStatus		Y	Valid values: 0 = New 1 = Partially Filled 9 = Suspended
1	Account			Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity			Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol		Q	OMNet short name.
48	Instrument/SecurityID		Q	Orderbook ID
22	Instrument/SecurityIDSource		Q	Valid values: M = Marketplace-assigned identifier
54	Side		Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty		Q	
40	OrdType		Q	Valid values: 2 = Limit
44	Price			
21101	UnderlyingColPx			Collateral Price - Required for limit orders in the Special Repo Markets.
59	TimeInForce		Q	Valid values: 1 = Good Till Cancel (GTC) 6 = Good Till Date (GTD)
432	ExpireDate			Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst			Valid values: G = All or None (AON)
151	LeavesQty		Y	

14	CumQty	Y	
6	AvgPx	Y	Always set to 0.0
60	TransactTime	Q	
378	ExecRestatementReason	Q	Valid values: 1 = GT renewal / restatement (no corporate action)
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
111	MaxFloor		For hidden orders. Contains currently shown quantity.
1089	MatchIncrement		Block Size
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
	Standard Trailer	Y	

6.10.15 Execution Report – Fill (out)

Purpose: Order Fill.

.

Identified by: MsgType = 8 AND ExecType = F

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	
11	ClOrdID		Q	Unique identifier set by the client.
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 = Giveup Clearing Firm 28 = Custodian 36 = Entering Trader 83 = Clearing Account
880	TrdMatchID		Q	Match ID assigned by the matching engine.
17	ExecID		Y	Unique identifier of execution message

150	ExecType	Y	Valid values: F = Trade
39	OrdStatus	Y	Valid values: 1 = Partially Filled 2 = Filled
1	Account		Optional pass-thru field set by client and echoed back by marketplace.
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	OMNet short name.
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit K = Market With Left Over as Limit (market order with unexecuted quantity becoming limit order at last price)
44	Price		
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD) 9 = At Crossing S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst		Valid values: G = All or None (AON)
32	LastQty	Q	Quantity (e.g. shares) bought/sold on this (last) fill.
31	LastPx	Q	Price of this (last) fill.
21100	LastUnderlyingPx		
151	LeavesQty	Y	Quantity open for further execution.
14	CumQty	Y	Currently executed quantity for chain of orders. **NOTE:** Will be 0 for fills on quotes.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.

797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
111	MaxFloor		For hidden orders. Contains currently shown quantity.
1089	MatchIncrement		Block Size
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.16 Execution Report – Order Suspended (out)

Purpose: Order Suspended (likely caused by temporary loss of connectivity).
Identified by: MsgType = 8 AND ExecType = 9

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	
11	ClOrdID	Q	Unique identifier set by the client.
453	NoPartyIDs		Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→ 448	PartyID	Q	Party identifier.
→ 447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→ 452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 28 = Custodian 36 = Entering Trader
17	ExecID	Y	
150	ExecType	Y	Valid values: 9 = Suspended
39	OrdStatus	Y	Valid values: 9 = Suspended
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	Short name of security
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier

54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit
44	Price		
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
1100	TriggeringInstruction/ TriggerType		Determines what should trigger an order modification. Valid values: 2 = Specified Trading Session 4 = Price Movement
1101	TriggeringInstruction/ TriggerAction		Defines the type of action to take when the trigger hits. Valid values: 1 = Activate
1102	TriggeringInstruction/ TriggerPrice		A specified limit price to validate against price movements –the trigger hits when the price is reached.
1113	TriggeringInstruction/ TriggerTradingSessionID		Identifier of Trading Session when the order is to be triggered.
1103	TriggeringInstruction/ TriggerSymbol		Symbol used for price triggers
1104	TriggeringInstruction/ TriggerSecurityID		Identifier of the security used for price triggers.
1105	TriggeringInstruction/ TriggerSecurityIDSource		SecurityIDSource of the instrument used for price triggering. Valid values: M = Marketplace-assigned identifier
1107	TriggeringInstruction/ TriggerPriceType		Determines what price should be tracked for price movements. Valid values: 1 = Best Offer 2 = Last Trade 3 = Best Bid
1109	TriggeringInstruction/ TriggerPriceDirection		Used to specify if the trigger should hit only on rising (Up) or falling (Down) prices. 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.
336	TradingSessionID		Valid Values: ‘A’ for Off-hour Orders.
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD) 9 = At Crossing

			S = Good till end of session (GTS)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
18	ExecInst		Valid values: G = All or None (AON)
151	LeavesQty	Y	Quantity open for further execution.
14	CumQty	Y	Currently executed quantity for chain of orders.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
797	CopyMsgIndicator		Set to ‘Y’ on Drop Copy messages
111	MaxFloor		For hidden orders. Contains currently shown quantity.
1089	MatchIncrement		Block Size
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer	Y	

6.10.17 Execution Report – Expired (out)

See Order Expiry section for details. If a GTD order is cancelled due to a Corporate Action the Expired Transaction will be sent out.

Identified by: MsgType = 8 AND ExecType = C

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	Genium INET order_number
11	ClOrdID	Q	Unique identifier set by the client.
453	NoPartyIDs		Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448 PartyID	Q	Party identifier.
→	447 PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452 PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 28 = Custodian 36 = Entering Trader

17	ExecID	Y	Identifier for this execution report. Integer value.
150	ExecType	Y	Valid values: C = Expired
39	OrdStatus	Y	Valid values: C = Expired
1	Account		Optional pass-thru field set by client and echoed back by marketplace. From Order
528	OrderCapacity		Account type. From Order **NOTE:** OrderCapacity is not used by derivatives market
55	Instrument/Symbol	Q	Short name of security
48	Instrument/SecurityID	Q	Orderbook ID
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Sell Short
38	OrderQtyData/OrderQty		Order quantity
40	OrdType	Q	Valid values: 2 = Limit
44	Price		Order price
21101	UnderlyingColPx		Collateral Price - Required for limit orders in the Special Repo Markets.
59	TimeInForce	Q	Valid values: 1 = Good Till Cancel (GTC) 6 = Good Till Date (GTD) S = Good till end of session (GTS)
18	ExecInst		Valid values: G = All or None (AON)
151	LeavesQty	Y	Will be 0 on expired orders.
14	CumQty	Y	
6	AvgPx	Y	Always set to 0.0
60	TransactTime	Q	
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
111	MaxFloor		For hidden orders. Contains currently shown quantity.
1089	MatchIncrement		Block Size
20199	OpenCloseIndicator		The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10

			for details.
	Standard Trailer	Y	

6.10.18 Business Message Reject (out)

Purpose: Business message reject.
Identified by: MsgType = j

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

7 Multileg Orders

7.1 Overview

A multileg security is made up of multiple securities that are traded atomically. Swaps, option strategies, futures spreads, are a few examples of multileg securities. The requirement that all legs be traded in the quantities that make up the multileg security is the important distinction between a multileg order and a list order.

Two generalized approaches to trading multileg securities are supported by FIX. The first approach involves a market maintaining multileg securities as separate products for which markets can be created. This “product approach” is often used in electronic trading systems. The second approach is to trade the multileg security as a group of separate securities.

The multileg order can be traded using one of the following FIX trading models. The first two models are variations on the multileg security as a separate tradable product. The last models permits trading of multileg securities in environments where the multileg securities are not productized.

Pre-defined Multileg Security Model

A.k.a. *Standard Combinations*. Marketplace-defined multileg securities made available for trading. In Genium INET, Standard Combination orders are treated exactly as single orders. To mimic this behavior, the FIX representation of entering a Standard Combination order is a normal New Order Single.

User-defined Multileg Security Model

A.k.a. *Tailor-Made Combinations (TMC)*. User-defined multileg securities made available for trading.

Strategy orders

A.k.a. *Non-Standard Combinations*. Multileg orders for combinations of security where a product is not defined or made available for others to trade.

***NOTE:** Strategy Orders are not supported in this solution.*

7.2 Multileg Order Features

Multileg orders are traded just like ordinary single orders, i.e. they;

- Have the same types of trading instructions, although the set of possibilities is limited.
- Use the same response messages, e.g. Execution Reports
- Are canceled using the Order Cancel Request or message
- Share the same type of workflows as New Order Single and Order Cancel Replace Request

Please see chapter 6, Order Management for information on aspects shared with single order messages.

7.2.1 Creating a Tailor-Made Combination Instrument

When trading a TMC the properties of each the legs are important. Each leg has the following properties:

- The instrument of the leg. This is represented by the LegSymbol (600) or LegSecurityID (602) fields.
- The Ratio Quantity of the leg. The relative number of contracts between the TMC legs. The FIX field to be used is LegRatioQuantity (623).
- The Side of each leg. The Side for each leg is relative to the TMC itself. The

The Security Definition Request is used to request creation of a TMC.

7.2.2 Multileg Order Limitations

Multileg orders have some limitations compared to regular orders. Most regular order features are available for multileg orders as well. The exceptions are:

- Overnight orders (TimeInForce= GTC or GTD) are not allowed.
- Reserve (Hidden) orders are not allowed.
- Triggers are not allowed.

7.3 Main Workflow

7.3.1 Submitting a Tailor-Made Combination Instrument Definition

A TMC is created by submitting a Security Definition Request to the marketplace. The system will respond with a Security Definition – TMC registration response (see section 0 for message details).

7.3.1.1 TMC Registration Response

The response to a submitted TMC registration request is a Security Definition message. This message will only contain the omnet series struct in integer format. *The actual instrument definition is only provided via reference data sessions, not via the FIX session where the registration was made.* The values in the FIX response can be used to identify the instrument definition in reference data. Using the instrument data received over omnet, the user can extract the omnet short name used as Symbol (55) in FIX to trade the instrument.

The SecurityResponseType (tag 323) will indicate whether the request was successful or not.

7.3.2 New Order

The multileg order workflow starts with user submitting an order.
In this solution, multileg orders are sent as ordinary New Order Single messages both for Standard Combination and Tailor-Made Combination Orders.
In response one Execution Report is produced for the multileg itself. The response will contain the OrderID that will be present in all later Execution Reports.

7.3.3 Order Modification

Order modification is accomplished using the Order Cancel Replace message. The message is used to modify an existing order and does not support delta updates (all relevant fields must be supplied). In response one Execution Report is produced for the multileg itself.

7.3.4 Multileg Status Reporting

Entering, cancelling or modifying an existing multileg order works exactly like any other instrument. Acknowledgements and rejects (Execution Report or Order Cancel Rejects) also look exactly like those for “ordinary” orders. See chapter 6 for details.
The only difference is with fills, which are sent per leg (see section 7.3.5).

NOTE:
A multileg order has a single OrderID (37) and ClOrdID (11), just like other orders. The legs are not considered to be orders in their own right.

7.3.5 Fills

When multileg orders are filled, Execution Reports are issued. The Execution Report – Combination Order Fill is used for multileg fills. See section 7.5.8 for message details.
Different models can be used in FIX to represent a fill. The model used in this solution is:

- **Multi-Leg only.** In this model a single or multiple Execution Report –Combination Order Fill is sent for the combination due to number of trade legs. If number of legs is smaller or equal to 15, single Execution Report –Combination Order Fill is sent. For more than 15 trade legs, another Execution Report –Combination Order Fill is sent. For example; 3 Execution Report –Combination Order Fill message is disseminated for 43 trade legs. The repeating group starting with the NoLegs (555) field (InstrmntLegExecGrp) will contain one entry per match that occurred in each leg. Each entry contains price and quantity.
  NOTE: It is entirely possible to receive more entries than the number of legs. There may have been more than one trade in each leg in a single matching round.

7.4 Workflows

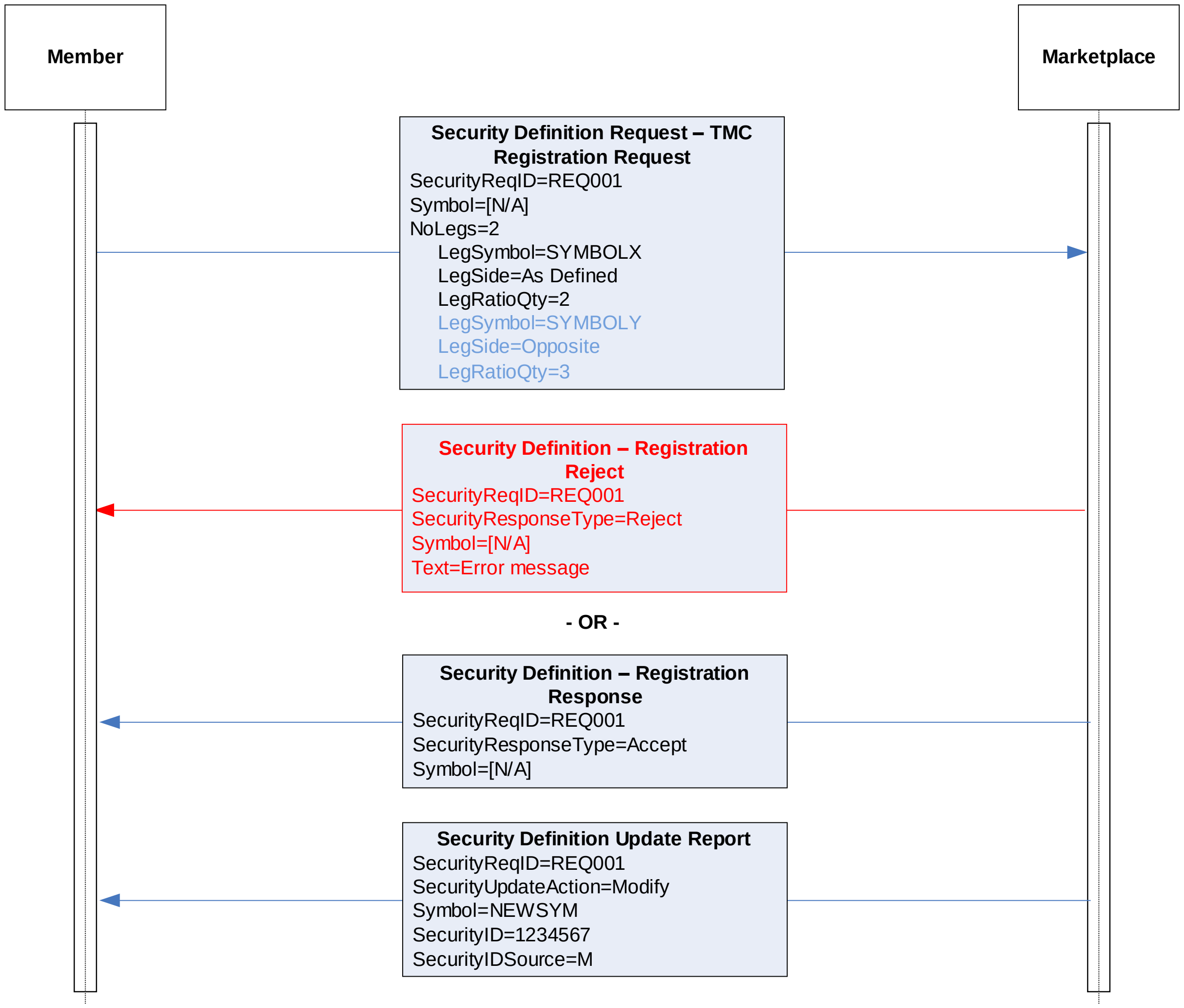
7.4.1 Registering a new TMC instrument

This example shows how to create a new Tailor-Made Combination with two legs. The two legs of requested TMC have the following properties:
Leg A (SYMBOLX):

- When a quantity of one (1) of the TMC is bought, a quantity of *2* (LegRatioQty=2) is *bought* (LegSide=As Defined).

Leg B (SYMBOLY):

- When a quantity of one (1) of the TMC is bought, a quantity of *3* (LegRatioQty=3) is *sold* (LegSide=Opposite).



Security Definition Request – TMC Registration Request
SecurityReqID=REQ001
Symbol=[N/A]
NoLegs=2
LegSymbol=SYMBOLX
LegSide=As Defined
LegRatioQty=2
LegSymbol=SYMBOLY
LegSide=Opposite
LegRatioQty=3

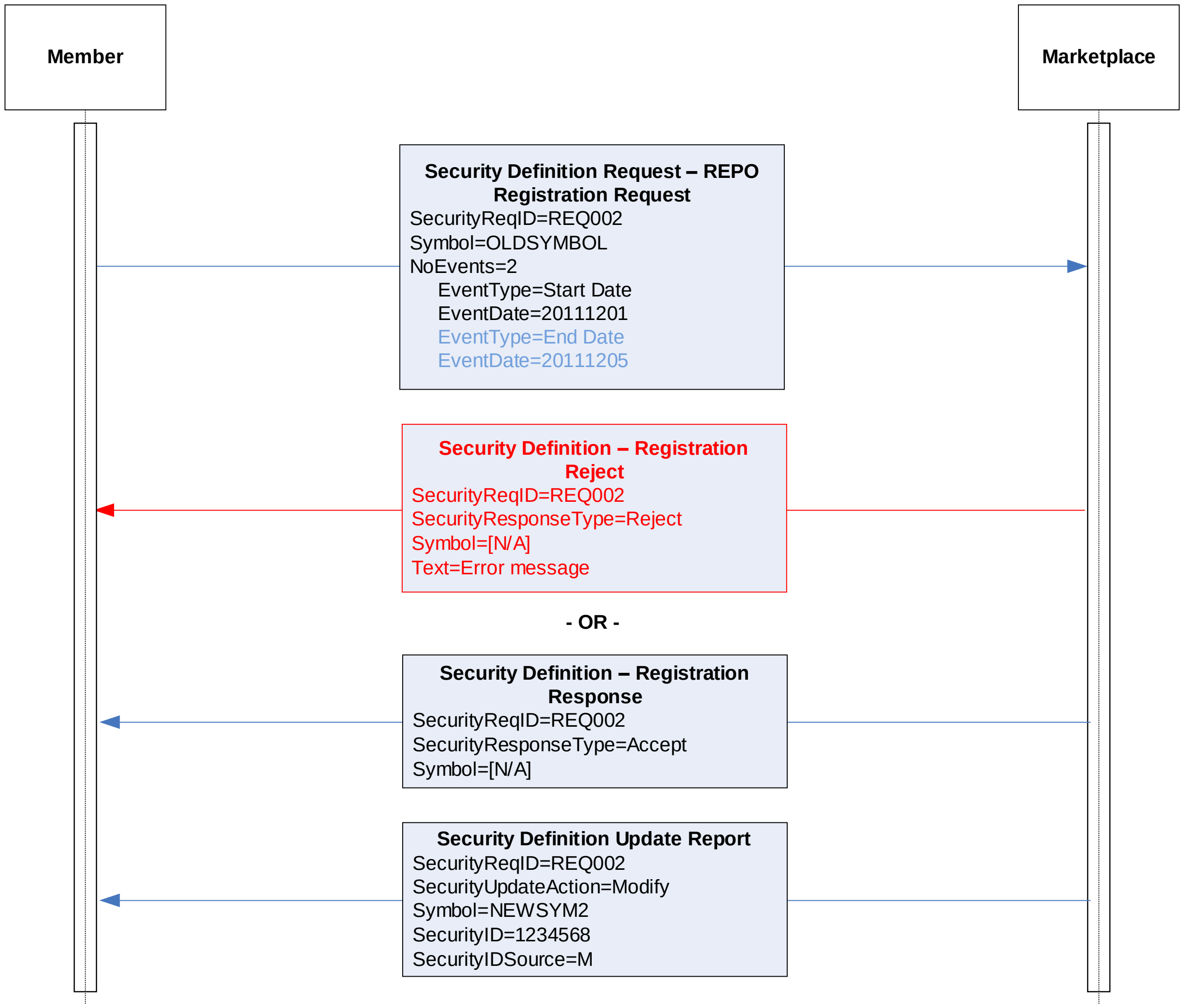
Security Definition – Registration Reject
SecurityReqID=REQ001
SecurityResponseType=Reject
Symbol=[N/A]
Text=Error message

Security Definition – Registration Response
SecurityReqID=REQ001
SecurityResponseType=Accept
Symbol=[N/A]

Security Definition Update Report
SecurityReqID=REQ001
SecurityUpdateAction=Modify
Symbol=NEWSYM
SecurityID=1234567
SecurityIDSource=M

7.4.2 Registering a new REPO instrument

This example shows how to register a new REPO instrument. The requested instrument has the same properties as OLDSYMBOL referred to in the request, but with new start and end dates.



Member

Marketplace

Security Definition Request – REPO Registration Request
SecurityReqID=REQ002
Symbol=OLDSYMBOL
NoEvents=2
EventType=Start Date
EventDate=20111201
EventType=End Date
EventDate=20111205

Security Definition – Registration Reject
SecurityReqID=REQ002
SecurityResponseType=Reject
Symbol=[N/A]
Text=Error message

- OR -

Security Definition – Registration Response
SecurityReqID=REQ002
SecurityResponseType=Accept
Symbol=[N/A]

Security Definition Update Report
SecurityReqID=REQ002
SecurityUpdateAction=Modify
Symbol=NEWSYM2
SecurityID=1234568
SecurityIDSource=M

7.5 Message Details

7.5.1 Security Definition Request –TMC Registration Request (in)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = c
320	SecurityReqID	Y	Client-generated identifier.
321	SecurityRequestType	Y	Type of Security Definition request. Valid values: 1 = Request Security identity for the specifications provided (name of the security is not supplied)
55	Instrument/Symbol	Y	Should be set to [N/A]
555	NoLegs	Q	Number of legs
→	600 LegSymbol	Q	OMNet short name for this leg. **NOTE:** if LegSecurityID+LegSecurityIDSource are

				used instead of LegSymbol, LegSymbol must be set to [N/A].
→	602	LegSecurityID		Orderbook ID for this leg.
→	603	LegSecurityIDSource		Valid values: M = Marketplace-assigned identifier

→	624	LegSide	Q	The side of this individual leg (multileg security).
				Valid values:
				B = As Defined
				C = Opposite

→	623	LegRatioQty	Q	The ratio of quantity for this individual leg relative to the entire multileg security.
	Standard Trailer		Y	

7.5.2 Security Definition Request –Repo Registration Request (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = c
320	SecurityReqID		Y	Client-generated identifier.
321	SecurityRequestType		Y	Type of Security Definition request. Valid values: 1 = Request Security identity for the specifications provided (name of the security is not supplied)
55	Symbol			Symbol or SecurityID+SecurityIDSource must be set to an existing repo instrument.
48	SecurityID			
22	SecurityIDSource			
864	NoEvents		Q	
→	865	EventType	Q	Valid values: 101 = Start Date (NASDAQ Extension) 102 = End Date (NASDAQ Extension)
→	866	EventDate	Q	Date of Event
	Standard Trailer		Y	

7.5.3 Security Definition Request – TM Precious Metal Registration Request (in)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = c
320	SecurityReqID	Y	Client-generated identifier.
321	SecurityRequestType	Y	Type of Security Definition request. Valid values: 1 = Request Security identity for the specifications provided (name of the security is not supplied)
55	Symbol		Symbol or SecurityID+SecurityIDSource must be set to an existing instrument.
48	SecurityID		
22	SecurityIDSource		
1147	UnitOfMeasureQuantity	Q	The weight of the metal, accurate to 5

				decimal places.
21060	Fineness		Q	The purity of the metal, accurate to 4 decimal places. Valid range: 0.0001 to 1.0000
864	NoEvents		Q	
→	865	EventType	Q	Valid values: 102 = End Date (NASDAQ Extension)
→	866	EventDate	Q	Date of Event
	Standard Trailer		Y	

7.5.4 Security Definition Request – Flexible Derivative Registration Request (in)

Tag	FIX tag name		Req’d	Comment
	Standard Header		Y	MsgType = c
320	SecurityReqID		Y	Client-generated identifier.
321	SecurityRequestType		Y	Type of Security Definition request. Valid values: 1 = Request Security identity for the specifications provided (name of the security is not supplied)
55	Symbol			Symbol or SecurityID+SecurityIDSource must be set to an existing instrument.
48	SecurityID			
22	SecurityIDSource			
202	StrikePrice			
864	NoEvents		Q	
→	865	EventType	Q	Valid values: 102 = End Date (NASDAQ Extension)
→	866	EventDate	Q	Date of Event
	Standard Trailer		Y	

7.5.5 Security Definition – Registration Response (out)

Purpose: Acceptance of a Security registration request.
Identified by: MsgType = d AND SecurityResponseType = 1

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = d
320	SecurityReqID	Y	Client-generated identifier.
323	SecurityResponseType	Q	Type of Security Definition message response. Valid values: 1 = Accept security proposal as-is
55	Instrument/Symbol	Q	Symbol not generated when this message is generated. Will be set to [N/A].
48	Instrument/SecurityID	Q	Contains the Omnet series struct in integer form with the fields separated by colons ":":

				country_c : market_c : instrument_group_c : modifier_c : commodity_n : expiration_date_n : strike_price_i
22	Instrument/SecurityIDSource		Q	101 = Genium INET series definition (NASDAQ Extension)
20013	DeferredPublicationTime			The number of minutes the publication of this trade will be delayed. The time is relative to time of agreement (TransBkdTime, tag 483).
1940	AssetType			Valid values: Precious
1938	AssetClass			Valid values: 5 – Commodity
1939	AssetSubClass			Valid values: 13 – Metal
996	UnitOfMeasure			Metal weight unit. The certified weight of the precious metal. Valid values: KG – kilogram GR – gram OZ – Troy ounce
21061	FinenessNotation			The market convention for notation of the metal purity
21062	MetalWeightConversionFactor			The conversion factor is used to normalize the order and trade price
21060	Fineness			The purity of the metal, accurate to 4 decimal places. Valid range: 0.0001 to 1.0000
1147	UnitOfMeasureQty			The certified weight of the precious metal
2304	NoAssetAttributes			
→	2305	AssetAttributeType		Valid values: MetalType Shape Refinery
→	2306	AssetAttributeValue		Asset attribute values directly depend on previously selected type. For metal types: AU – gold AG – silver PT – platinum PD – palladium For shape types: BA – Bar MB – Mini Bar LB – Large Bar GB – Granule Bag NS – Non-standard OR – Ore For refinery types: I – GDL In O – GDL Out
	Standard Trailer		Y	

7.5.6 Security Definition –Registration Reject (out)

Purpose: Reject of a Security registration request.
Identified by: MsgType = d AND SecurityResponseType = 5

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = d
320	SecurityReqID	Y	Client-generated identifier.
323	SecurityResponseType	Q	Type of Security Definition message response. Valid values: 5 = Reject security proposal
58	Text	Q	Error message
	Standard Trailer	Y	

7.5.7 Security Definition Update Report (out)

Purpose: Return instrument identifiers usable for FIX.
Identified by: MsgType = BP

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = BP
320	SecurityReqID		Y	Client-generated identifier.
980	SecurityUpdateAction		Q	Valid values: M = Modify
55	Symbol		Q	Symbol of created instrument.
48	SecurityID		Q	Order book ID of created instrument.
22	SecurityIDSource		Q	Valid values: M = Marketplace-assigned identifier
454	NoSecurityAltID		Q	Number of alternate SecurityIDs. Will always be 1.
→	455	SecurityAltID	Q	Contains the Omnet series struct in integer form with the fields separated by colons ":": country_c : market_c : instrument_group_c : modifier_c : commodity_n : expiration_date_n : strike_price_i
→	456	SecurityAltIDSource	Q	101 = Genium INET series definition (NASDAQ Extension)
555	NoLegs			Number of legs (for strategy/combination) instruments.
→	600	LegSymbol		Short name of leg instrument.
→	602	LegSecurityID		Order book ID of leg instrument.
→	603	LegSecurityIDSource		Valid values: M = Marketplace-assigned identifier
→	623	LegRatioQty		The ratio of quantity for this individual leg relative to the entire multileg security.

→	624	LegSide		The side of this individual leg (multileg security). Valid values:
				B = As Defined
				C = Opposite

1940	AssetType			Valid values:

				Precious
1938	AssetClass			Valid values: 5 – Commodity
1939	AssetSubClass			Valid values: 13 – Metal
996	UnitOfMeasure			Metal weight unit. The certified weight of the precious metal. Valid values: KG – kilogram GR – gram OZ – Troy ounce
21061	FinenessNotation			The market convention for notation of the metal purity
21062	MetalWeightConversionFactor			The conversion factor is used to normalize the order and trade price
21060	Fineness			The purity of the metal, accurate to 4 decimal places. Valid range: 0.0001 to 1.0000
1147	UnitOfMeasureQty			The certified weight of the precious metal
2304	NoAssetAttributes			
→	2305	AssetAttributeType		Valid values: MetalType Shape Refinery
→	2306	AssetAttributeValue		Asset attribute values directly depend on previously selected type. For metal types: AU – gold AG – silver PT – platinum PD – palladium For shape types: BA – Bar MB – Mini Bar LB – Large Bar GB – Granule Bag NS – Non-standard OR – Ore For refinery types: I – GDL In O – GDL Out
	Standard Trailer		Y	

7.5.8 Execution Report – Combination Order Fill (out)

Purpose: Combination Order Fill.
Identified by: MsgType = 8 AND ExecType = F AND MultiLegReportingType = 3

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = 8
37	OrderID	Y	

11	ClOrdID		Q	
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional Clearing Firm and Clearing Account.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 4 = Clearing Firm 7 = Entering Firm 12 = Executing Trader 36 = Entering Trader 83 = Clearing Account
880	TrdMatchID		Q	Match ID assigned by the matching engine.
20034	ComboMatchID		Q	Combination Match ID assigned by the matching engine.
17	ExecID		Y	
150	ExecType		Y	Valid values: F = Trade
39	OrdStatus		Y	Valid values: 1 = Partially filled 2 = Filled
1	Account			Optional pass-thru field set by client and echoed back by marketplace.
55	Instrument/Symbol		Q	Combination orderbook OMNet short name.
48	Instrument/SecurityID		Q	Combination Orderbook ID
22	Instrument/SecurityIDSource		Q	Valid values: M = Marketplace-assigned identifier
54	Side		Y	Valid values: 1 = Buy 2 = Sell
40	OrdType		Q	Valid values: 1 = Market 2 = Limit
44	Price		Q	Net price of the combination as entered in the order.
59	TimeInForce		Q	Valid values: 0 = Day 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) S = NASDAQ Extension: Good till End of Session (GTS)
336	TradingSessionID			State type of order expiration. Conditionally required if TimeInForce = GTS. Valid Values: StateTypeId for GTS or ‘A’ for Off-hour Orders.
32	LastQty			Quantity (e.g. shares) bought/sold on this (last) fill.
31	LastPx			Net price of this (last) multileg fill.
151	LeavesQty		Y	

14	CumQty		Y	
6	AvgPx		Y	Note: Always set to 0.0
60	TransactTime		Y	
442	MultiLegReportingType		Q	Valid values: 3 = Multi-leg security
555	NoLegs		Q	Number of legs involved in execution. Maximum number of legs in a message is 15.
→	600	LegSymbol	Q	Omnet short name of leg security
→	602	LegSecurityID	Q	Orderbook ID of leg security
→	603	LegSecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
→	637	LegLastPx	Q	Trade price for this leg
→	1418	LegLastQty	Q	NASDAQ Extension: Quantity traded in this leg
→	20200	LegTrdMatchID	Q	Indicates same value with 880(TrdMatchID) tag of the matched order
70	AllocID			NASDAQ Extension: Optional pass-thru field set by client and echoed back by marketplace.
797	CopyMsgIndicator			Set to ‘Y’ on Drop Copy messages
20199	OpenCloseIndicator			The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close 4 = Set to default for account Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer			

8 General Quote Handling

8.1 Introduction

The Mass Quote message is used by market makers and other actors with similar responsibilities to send quotes into a market. The quote messages, as described in this section, are typically used to send continuous unsolicited quotes in markets with tradable quoting. Such quotes are sent by quote issuers (market makers, specialists, liquidity providers or similar), i.e. actors that have an obligation to provide continuous liquidity in the market.

A quote is two-sided, i.e. normally contains both bid and offer price and size. Many marketplaces limit market makers to one (two-sided) quote in each security.

The Mass Quote message allows the user to submit multiple quotes in a single message.

Responses (acks / rejects) for Mass Quote messages are subject to bilateral agreement between parties and/or as specified in the QuoteResponseLevel (301) field of the request message.

NOTE: The Time-in-Force for continuous quotes is considered to be *Day* in this solution.

8.2 Solution restrictions

Quoting in FIX for Genium INET has the following restrictions:

- Only one two-sided quote per actor per instrument is allowed. This simplifies quote cancellation and generation of quote identifiers, see below.
- The response to a Mass Quote is restricted to negative acks (QuoteResponseLevel = 1).
- All quotes are assumed to be valid until end of day (or until canceled).
- Replacing a quote is as simple as sending a new Mass Quote for the same instrument(s).
- Cancel of a mass quote is achieved by sending a new mass quote with all prices and quantities to 0 (see section Quote Cancellation).
- The pass-thru fields (Account and AllocID) supported in order entry and trade reporting, are **not** supported in quoting transactions.

8.3 Quote Modification

Quote modification is accomplished through the use of the same messages as when adding a quote, i.e. through the Mass Quote message. Replacing a quote in a single quote market is straightforward as every update replaces the old one based on the quote issuer, security (series) and side.
It is possible to replace a double sided quote and leave one side unchanged (for example to avoid loosing priority).
To leave one side of the quote unchanged, set the quantity (Bid or Offer size) on that side set to 0. The price on that side must be set to a non-zero value. This is true even if the currently quoted price is zero (zero is a valid price for certain instruments). If both price and quantity is set to zero that will be interpreted as a quote cancel (see below).

8.4 Quote Cancellation

A quote can be canceled (or withdrawn) by sending a Mass Quote message with bid and offer prices and sizes all set to zero:

- BidPx (132) = 0
- OfferPx (133) = 0
- BidSize (134) = 0
- OfferSize (135) = 0

It is possible to cancel only one side of a double-sided quote by setting the price and the quantity on that side to zero.

8.5 Quote Request

Any participant can issue a quote request message to request market makers to enter quotes in an order book. The quote Request is sent either to all market makers or to the list of specified market makers. Information about available market makers for an instrument is sent out through the Fix Reference Data Server in the Security Definition and Security Definition Update messages.

A Quote Request must contain:

- The instrument a quote is requested for.

It may also optionally contain:

- A minimum quantity

- A side (if not set a request for double-sided quote is assumed)
- List of market makers to send the request to

For Derivatives Market, it is possible to send quote request not only to market makers but also to all market participants.

8.6 Main Workflow

8.6.1 Mass Quotes

The Mass Quote message can contain quotes for multiple securities to support applications that allow for the mass quoting of e.g. 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).

A QuoteSet specifies the first level of repeating fields for the Mass Quote message. It represents a group of related quotes and can, for example, represent an option class.

Each QuoteSet contains a repeating group of QuoteEntries where each entry represents an individual two-sided quote.

NOTE: This flexible construct is not fully supported in this implementation. We limit each Mass Quote to contain a single Quote Set. The number of quote entries supported is limited by the back-end. See note below.

It is possible that the number of Quote Entries for a Quote Set could exceed one’s physical or practical message size. It may be necessary to fragment a message across multiple quote messages.

NOTE: The maximum number of quotes in a Mass Quote of *tradable quotes* is configured in the back-end system. See relevant documentation.

The grouping of quotes is as follows:

- NoQuoteSets – specifies the number of sets of quotes contained in the message. Will always be one in this solution.
    - QuoteSetID – Is a unique ID given to the quote set within the message. Required in FIX. Will be ignored by the back-end.
    - TotQuoteEntries – defines the number of quotes for the quote set across all messages
    - NoQuoteEntries – defines the number of quotes contained within this message for this quote set
        - QuoteEntryID – Is a unique ID given to a specific quote entry. Can be set to 1, since only one quote per instrument is allowed.
        - Information regarding the security/book to which the quote belong
        - Information regarding the specific quote (bid/ask size and price).

NOTE: It is strongly recommended to set the QuoteEntryIDs as an increasing number starting from 1 on the first entry in each Mass Quote message. This enables the quote issuer to easily identify what quotes have been rejected in case that happens.

8.6.1.1 Limitations

The Mass Quote message can be populated with quotes for different securities as long as they belong to the same partition in Genium INET. Please see relevant Genium INET documentation for information on how to tell which partition a security belongs.

8.6.2 Mass Quote Acknowledgement

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]). The ability to reject an individual quote entry is important so that the majority of quotes can be successfully applied to the market instead of having to reject the entire Mass Quote for a minority of rejected quotes. The Mass Quote Ack has two uses:

- Some quotes rejected. See section 8.11.2 for message details.
- All quotes rejected. See 8.11.3 for message details.

8.6.3 Quote Rejects

The Mass Quote Acknowledgement message – is primarily used to reject Mass Quotes.

8.6.4 Quote Request

The Quote Request message is used to request for quotes in a specific order book. A Quote Request Reject is returned by the system if the Quote Request is rejected.

8.7 Quote Identifiers

8.7.1 Quote Message identifier

Every inbound quote message must be associated with a unique message identifier per FIX session. The message identifier can be used to keep an audit trail of quote updates and is used to link a request message to responses. The message identifier is echoed back on response, fill and other messages that are sent out based on a quote.

- **Quote ID**

  The QuoteID (117) is the message identifier used in Mass Quote messages.

The message identifier is relayed back in the following messages:

Request Message	Response Message	Message Identifier Mapping
Mass Quote	Mass Quote Acknowledgement	MQ.QuoteID → MQA.QuoteID
N/A	Execution Report,	MQ.QuoteID → ClOrdID

When alternative fields (“or”) are shown in the table, the field to use depends on what message was last used to update the quote.

NOTE: It is **strongly** recommended that the QuoteIDs are taken from the same numbering series as the ClOrdID in cases where Orders and Quotes are submitted through the same FIX session. Quote issuers using multiple sessions or even trading applications should ensure QuoteID uniqueness.

8.7.2 Quote Entity Identifier

Every quote must be associated with a unique entity identifier. The identifier is used to identify an individual quote when updating quotes. The identifier can be compared to the OrderID (37) of orders, but is normally static over time as the same quote is continuously updated. Another difference from the OrderID is that the quote issuer includes the identifier in the Quote messages, it is not returned by the marketplace in responses to quotes.

- Quote Entry ID

  The QuoteEntryID (299) is the entity identifier used in Mass Quote messages. Since only a single quote is allowed per orderbook and side per issuer, there is no strict FIX requirement to set this to a unique value. However, when the back-end rejects a particular quote in a Mass Quote, it only returns the number of the quote entry counting from the first entry in the Mass Quote. So it is strongly recommended to adopt the same numbering scheme for QuoteEntryIDs; *Number the first entry in the Mass Quote 1, the following 2 etc. This way it will be easy to identify rejected entries*.

It should be noted that a quote issuer is never allowed to have more than one two-sided quote in a single book – irrespective of what identifiers are used.

The Quote entity ID (QuoteEntryID) is echoed back in the following messages:

Request Message	Response Message	Quote Entity Identifier Mapping
Mass Quote	Mass Quote Acknowledgement	MQ.QuoteEntryID → MQA.QuoteEntryID
N/A	Execution Report,	MQ.QuoteID → ClOrdID

When alternative fields ("or") are shown in the table, the field to use depends on what message was last used to update the quote.

8.8 Quote Response Level

Derivative markets are characterized by high bandwidth consumption – due to a change in an underlying security price causing multiple (often in the hundreds) of quotes to be recalculated and retransmitted to the market. For that reason the ability for market participants (and the market) to be able to set the level of response requested for a Mass Quote message is specified using the QuoteResponseLevel (301) field.

For *regular Mass Quotes* the only supported value is:

- 1 = Requests acknowledgement of invalid or erroneous quote messages only (negative)

8.9 Quote State Changes

A quote is, in principle, regarded as a permanent representation of interest from the relevant market maker, even though it may not always be externally visible (or implemented in the trading engine). This means the quote always has a state: it can only be "not found" when the market maker does not have any side quoted for a security. Empty (or "zero") quote sides are represented using BidPx, OfferPx, BidSize and OfferSize

= 0. However, zero quotes can also be implemented as non-existing quotes.

Quote state changes are divulged by:

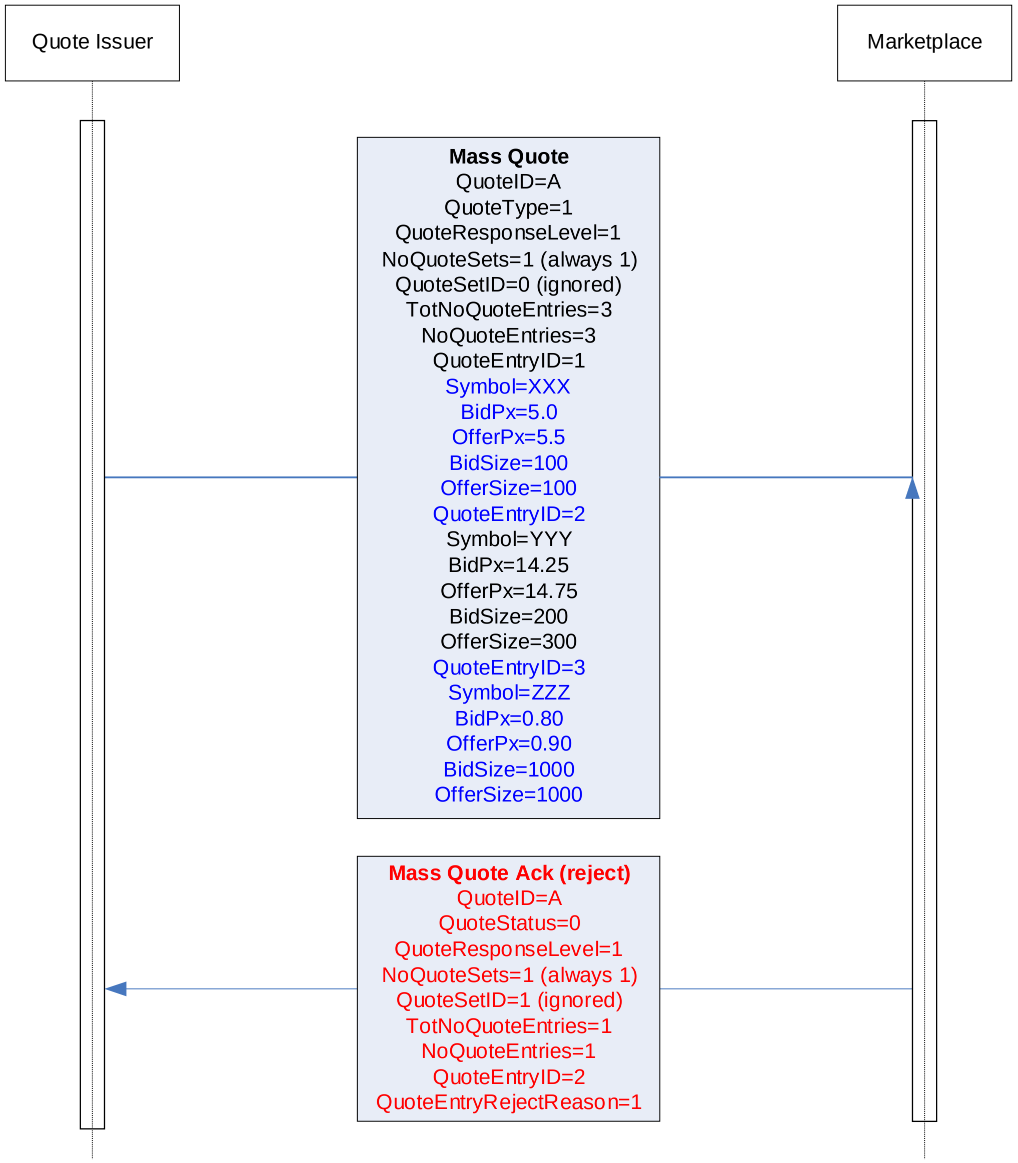
- The Mass Quote Acknowledgement message after a quote update (excluding fills) and subject to the specified or bilaterally agreed QuoteReponseLevel
- Execution Reports after fills

A quote can move from any state to any other state.

8.10 Workflows

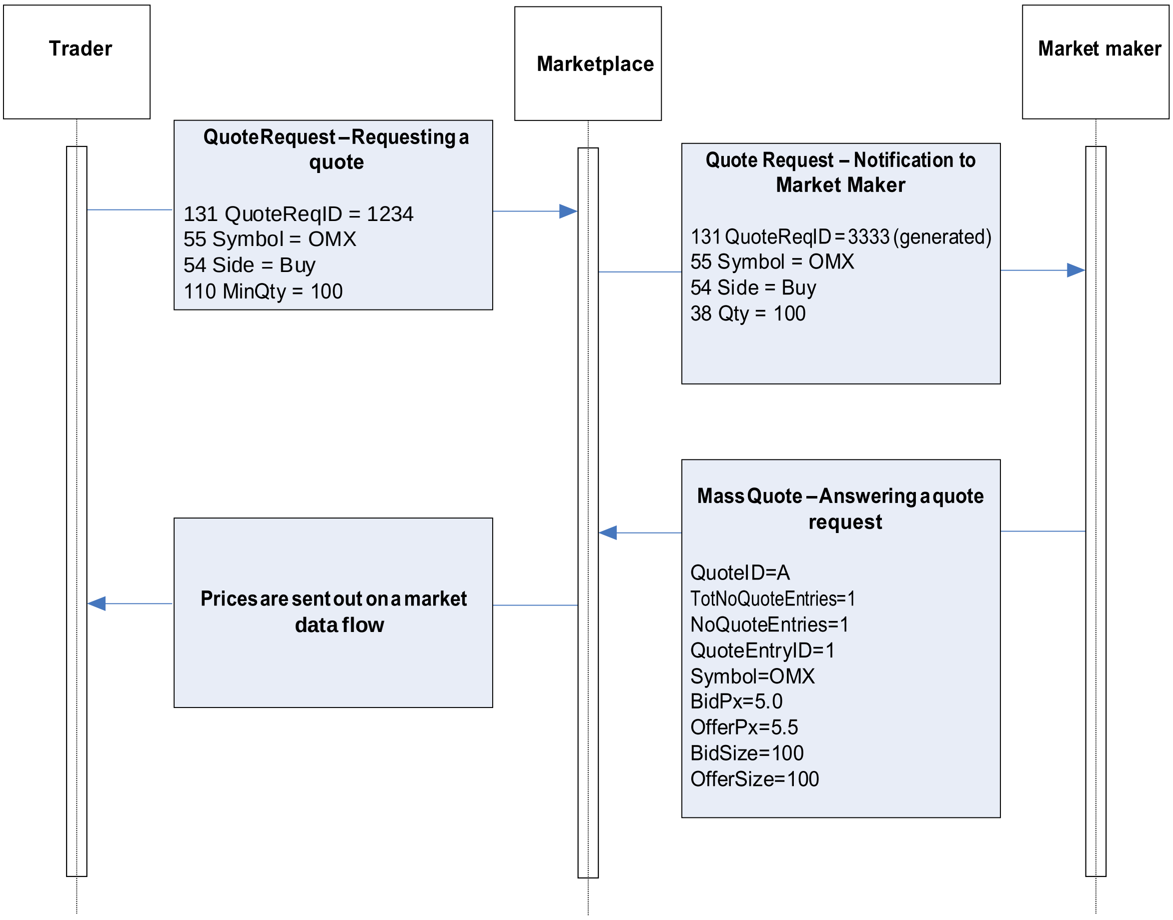
8.10.1 Mass Quote with individual reject

In this scenario a Mass Quote with three entries is sent in. The second entry in the Mass Quote is rejected. Notice how the second entry is numbered 2 (QuoteEntryID=2) in the reject. This will be true regardless how QuoteEntryIDs are set in the inbound Mass Quote. So it is strongly advised to set the QuoteEntryIDs similarlyon the inbound Mass Quotes.



8.10.2 Qute Request Scenario

In this scenario a Quote Request is sent in from a Trader. The Quote Request is then sent to the Market Maker who answers the Quote Request with a Mass Quote.



8.11 Message Details

8.11.1 Mass Quote (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = i
117	QuoteID		Y	Quote issuer assigned message identifier
537	QuoteType		Q	Identifies the type of quote. Valid values: 1 = Tradeable
301	QuoteResponseLevel		Q	Level of Response requested from receiver of quote messages. Valid values: 1 = Acknowledge only negative or erroneous quotes
453	NoPartyIDs			Optional repeating group used for on behalf of transactions and/or for optional AFK/Custodian identifier.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values:

					D = Proprietary/Custom code
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified. Valid values: 28 = Custodian
1	Account			Q	Requiredl pass-thru field set by client and echoed back by marketplace.
528	OrderCapacity			Q	Account type Valid values: A = Customer P = Principal F = Fund Order C = Primary Dealer Corp K = Primary Dealer Govt O = Portfolio Mgmt Company T = Investment Trust Y = Other Fund S = Special Client N = Other 2 D = Other 3 **NOTE:** OrderCapacity is not used by derivatives market
296	NoQuoteSets			Y	Only one Quote set allowed in this solution.
→	302	QuoteSetID		Y	Required in FIX. Will be ignored by the back-end.
→	304	TotNoQuoteEntries		Y	Total number of quotes for all quote sets (will be equal to NoQuoteEntries in this solution).
→	295	NoQuoteEntries		Y	Number of double-sided quotes in Quote Set.
→	→	299	QuoteEntryID	Y	Recommended to be set to an increasing number, starting with 1 in each Mass Quote.
→	→	55	Instrument/Symbol		OMNet short name. Symbol or SecurityID+SecurityIDSource must be set.
→	→	48	Instrument/Security ID		Orderbook ID
→	→	22	Instrument/Security IDSource		Valid values: M = Marketplace-assigned identifier
→	→	132	BidPx	Q	
→	→	133	OfferPx	Q	
→	→	134	BidSize	Q	
→	→	135	OfferSize	Q	
	Standard Trailer			Y	

8.11.2 Mass Quote Acknowledgement – some quotes rejected (out)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = b
117	QuoteID		
297	QuoteStatus	Y	Identifies the status of the mass quote acknowledgement. Valid values: 0 = Accept

301	QuoteResponseLevel			Q	Level of Response requested from receiver of quote messages. Valid values: 1 = Acknowledge only negative or erroneous quotes
537	QuoteType			Q	Identifies the type of quote. Valid values:1 = Tradeable
453	NoPartyIDs				Optional repeating group used for on behalf of transactions.
→	448	PartyID		Q	Party identifier.
→	447	PartyIDSource		Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 36 = Entering Trader
296	QuoteSetAckGrp/NoQuoteSets			Y	Multiple quote sets not supported. Will always be 1.
→	302	QuoteSetID		Y	Required in FIX. Will be set to 1.
→	295	NoQuoteEntries		Y	Number of double-sided quotes in Quote Set.
→	→	299	QuoteEntryID	Y	Will be set to the number in the order the entries appeared in the incoming Mass Quote (regardless of the QuoteEntryIDs actually set in the Mass Quote). Example: Will be set to 2 if the second entry in the Mass Quote was rejected.
→	→	368	QuoteEntryRejectReason		Reject reason for this individual quote.
58	Text				Will contain the error message(s) from the back-end. **NOTE:** If more than one quote is rejected, the error messages are separated by a “#”.
	Standard Trailer			Y	

8.11.3 Mass Quote Acknowledgement – All Quotes Rejected (out)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = b
117	QuoteID	Q	From Mass Quote
297	QuoteStatus	Y	Identifies the status of the mass quote acknowledgement. Valid values: 5 = Reject
300	QuoteRejectReason		Reason Quote was rejected. Valid values: 6 = Duplicate Quote IDs 99 = Other
301	QuoteResponseLevel	Q	Level of Response requested from receiver of quote messages. Valid values: 1 = Acknowledge only negative or erroneous quotes
537	QuoteType	Q	Identifies the type of quote. Valid values:

				1 = Tradeable
453	NoPartyIDs			Optional repeating group used for on behalf of transactions.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 36 = Entering Trader
58	Text			
	Standard Trailer		Y	

8.11.4 Quote Request – (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = R
131	QuoteReqID		Q	Unique identifier assigned by the requestor. Will be returned in responses.
146	NoRelatedSym		Q	Will be set to 1
→	55	Intrumetn/Symbol	Y	Short name of instrument. **NOTE:** Must be set to [N/A] if SecurityID is used as instrument identifier.
→	48	Instrument/SecurityID		OrderBook ID
→	22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
→	54	Side		Valid values: 1 = Buy 2 = Sell **NOTE:** The absence of a side implies that a two-sided quote is being requested.
→	110	MinQty		Minimum quantity requested. **NOTE:** The absence of MinQty implies that any size is acceptable.
1116	NoRootParyIDs			Optional if sending to specified market makers
→	1117	RootPartyID	Q	Party identifier.
	1118	RootPartyIDSource	Q	Valid values: D = Proprietary/Custom code
	1119	RootPartyRole	Q	Role of the current PartyID. Valid values: 66 = Market Maker*
	Standard Trailer		Y	

* In existing Quote Request functionality, role of the counterparty may not always be Market Maker for Derivatives Market.

8.11.5 Quote Request – (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = R

131	QuoteReqID		Q	From Incoming Quote Request
1116	NoRootPartyIDs			Optional repeating group. Identifies the submitter of the Quote Request
→	1117	RootPartyID	Q	Party identifier.
→	1118	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	1119	RootPartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 7 = Entering Firm 36 = Entering Trader
146	NoRelatedSym		Q	Will be set to 1
→	55	Symbol	Q	Short name of Instrument
→	54	Side		Valid values: 1 = Buy 2 = Sell **NOTE:** The absence of a side implies that a two-sided quote is being requested.
→	110	MinQty		The minimum Quantity to fulfill
	Standard Trailer		Y	

8.11.6 Quote Request Reject (out)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = AG
131	QuoteReqID		Y	Unique identifier assigned by the requestor
658	QuoteRequstRejectReason		Y	Valid values: 99 = Other
146	NoRelatedSym		Q	Will be set to 1
→	55	Intrumetn/Symbol	Y	Short name of instrument. **NOTE:** Must be set to [N/A] if SecurityID is used as instrument identifier.
→	48	Instrument/SecurityID	Q	OrderBook ID
→	22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
58	Text			Free text description of the reject
	Standard Trailer		Y	

9 Reporting of privately negotiated Trades

9.1 Introduction

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

- Trades negotiated between market participants without using execution mechanisms provided by the Marketplace
- Trades formed at other execution venues but reported to the marketplace for regulatory or publication reasons. Such execution venues may include (systematic) internalizers, ECN's, ATS's, MTF's and others regulated markets. *(Not supported in this solution)*

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

One-Party Report for Matching

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

Two-Party Reports

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

9.2 Identifiers

9.2.1 Trade Report ID

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

9.2.2 Trade Report Reference ID

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

9.2.3 Secondary Trade Report ID

This ID (818) is set by the marketplace on Trade Capture Report Ack messages. It is an interim identifier assigned to the trade that is valid until the trade is confirmed. The Secondary Trade Report ID carries the Genium INET order_number. Analogous to the OrderID on Orders, this is the preferred identifier to use when canceling a previous Trade Capture Report since it requires no lookup in the gateway. To use it in a Trade Cancel, set SecondaryTradeReportRefID (881) to the value received in 818 in the previous TCR Ack message.

9.2.4 Secondary Trade Report Reference ID

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

9.2.5 Timestamps

9.2.5.1 Settlement Date

SettlDate (64) contains the Settlement date.

9.2.5.2 Time of Agreement

Time of agreement is shown by the TransBkdTime (483) field.

9.3 Main Workflow

9.3.1 Trade Capture Report

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

- To submit a new Trade Report (one-party or two-party)

- To update a Trade Report (not supported in this solution)
- To cancel a Trade Report
- For the marketplace to publish trade confirmations (see chapter 10)
- For the marketplace to publish updates to previous trade confirmations (see chapter 10)
- To cancel a confirmed trade (see chapter 10)
- For the marketplace to notify the contra party when a one-party report has been sent in.

9.3.1.1 Submitting a new Trade Report

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

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

.

9.3.1.2 Trade Types

The TrdType tag (828) is used to specify the type of trade being reported to the marketplace.
A list of supported Trade Type values is supplied separately by the marketplace. Note that there may be limitations on which trade types are allowed for a certain instrument and/or participant. It is out of scope of this document to fully specify all such rules. Please refer to the member trading rules for further information.

9.3.1.3 Marketplace notification to counterparty

When a one-party report for matching is first entered, the marketplace will send a TCR as a notification to the counterparty. See section 9.5.5 for message details.
If the counterparty cancels the trade report causing the notification, a Delete Notification to counterparty will be sent out. See section 9.5.6 for message details.
When the trade report causing the notification gets matched, a Delete Notification to counterparty will be sent out. See workflow example 10.3.2 for details.
NOTE: Notification to counterparty messages will not contain the TargetSubID (57). See section 10.1.5 for details.

9.3.1.4 Marketplace publication of Confirmed Trades

The marketplace uses the TCR to publish confirmed trades, whether auto-matched or reported by clients. See chapter 10 for details.

9.3.2 Trade Capture Report Acknowledgement

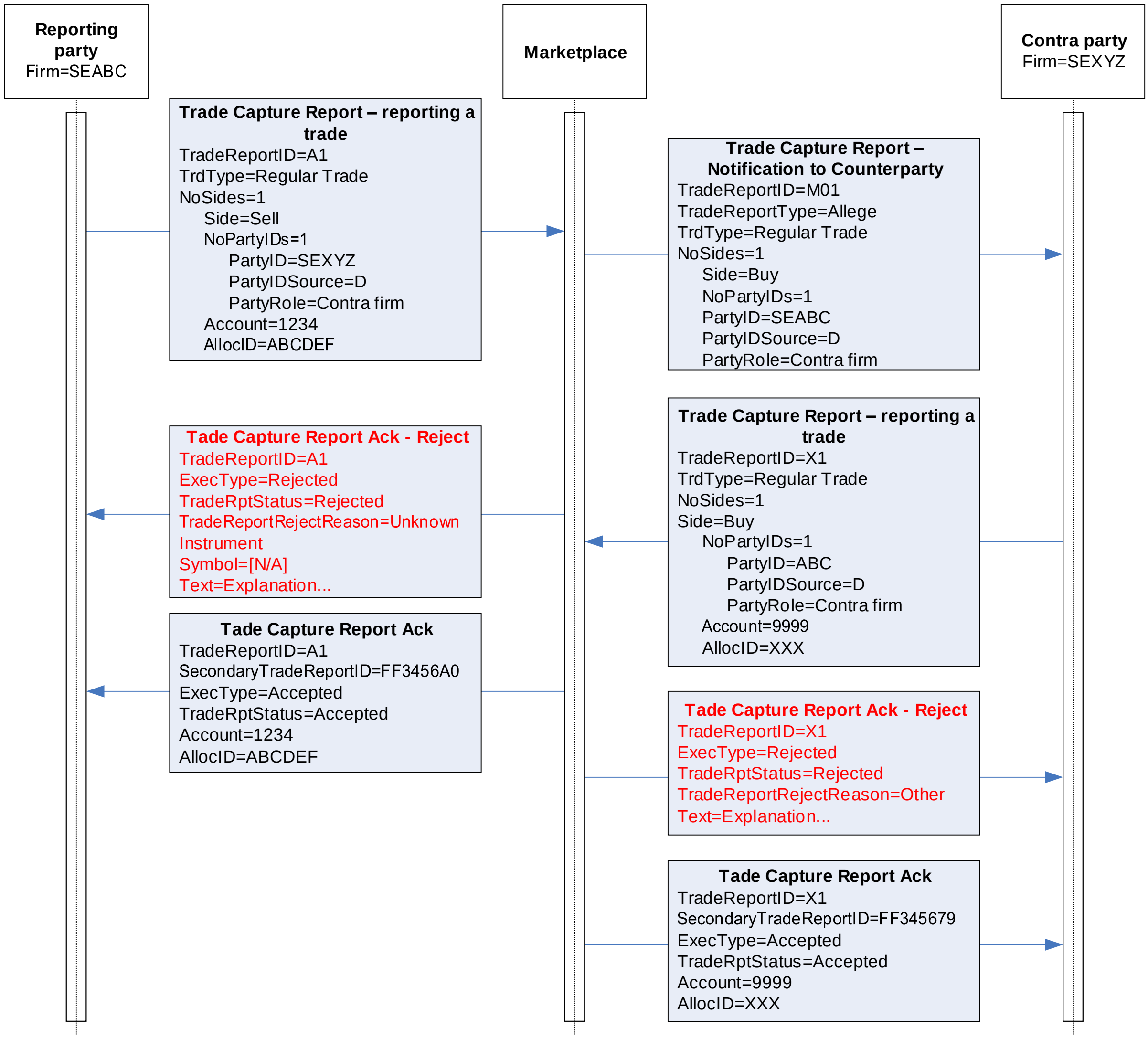
The TCR Ack is used to acknowledge or reject a Trade Capture Report submitted to the exchange. You will always receive a Trace Capture Report Ack when reporting a trade.

9.4 Workflows

The workflows presented here are meant to clarify the use of the most important fields in the Trade Capture Report and Trade Capture Report Ack messages. The workflows are based on the tables in FIX Protocol Specification 5.0 SP1 [1], Volume 5, Appendix B. They have been modified for this solution.

9.4.1 One-party Report for matching

In this example Each Side of a trade reports its own side. When the first party has reported his side, the counterparty receives a notification (see section 9.5.5 for message details).
When both sides have reported the trade it is matched. The resulting confirmation sent out to both reporting parties is described in detail in chapter 10.



Trade Capture Report – reporting a trade
TradeReportID=A1
TrdType=Regular Trade
NoSides=1
Side=Sell
NoPartyIDs=1
PartyID=SEXYZ
PartyIDSource=D
PartyRole=Contra firm
Account=1234
AllocID=ABCDEF

Trade Capture Report – Notification to Counterparty
TradeReportID=M01
TradeReportType=Allege
TrdType=Regular Trade
NoSides=1
Side=Buy
NoPartyIDs=1
PartyID=SEABC
PartyIDSource=D
PartyRole=Contra firm

Trade Capture Report – reporting a trade
TradeReportID=X1
TrdType=Regular Trade
NoSides=1
Side=Buy
NoPartyIDs=1
PartyID=ABC
PartyIDSource=D
PartyRole=Contra firm
Account=9999
AllocID=XXX

Tade Capture Report Ack - Reject
TradeReportID=A1
ExecType=Rejected
TradeRptStatus=Rejected
TradeReportRejectReason=Unknown Instrument
Symbol=[N/A]
Text=Explanation...

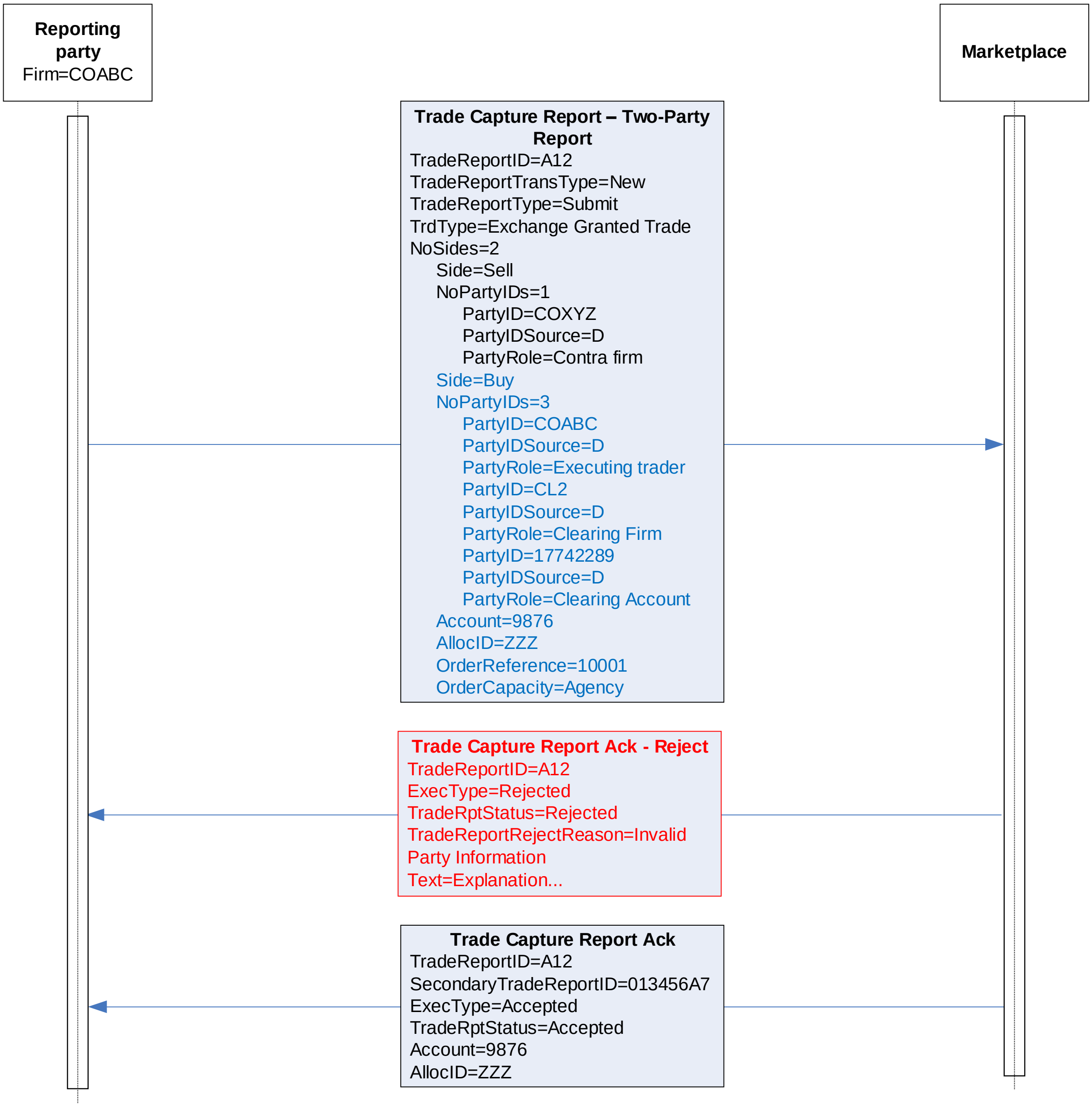
Tade Capture Report Ack
TradeReportID=A1
SecondaryTradeReportID=FF3456A0
ExecType=Accepted
TradeRptStatus=Accepted
Account=1234
AllocID=ABCDEF

Tade Capture Report Ack - Reject
TradeReportID=X1
ExecType=Rejected
TradeRptStatus=Rejected
TradeReportRejectReason=Other
Text=Explanation...

Tade Capture Report Ack
TradeReportID=X1
SecondaryTradeReportID=FF345679
ExecType=Accepted
TradeRptStatus=Accepted
Account=9999
AllocID=XXX

9.4.2 Two-party Report

The reporting party reports for both sides. When the Two-Party report is accepted a confirmation will be sent out to both parties. See chapter 10 for details.



Trade Capture Report – Two-Party Report

```
TradeReportID=A12
TradeReportTransType=New
TradeReportType=Submit
TrdType=Exchange Granted Trade
NoSides=2
    Side=Sell
    NoPartyIDs=1
        PartyID=COXYZ
        PartyIDSource=D
        PartyRole=Contra firm
    Side=Buy
    NoPartyIDs=3
        PartyID=COABC
        PartyIDSource=D
        PartyRole=Executing trader
        PartyID=CL2
        PartyIDSource=D
        PartyRole=Clearing Firm
        PartyID=17742289
        PartyIDSource=D
        PartyRole=Clearing Account
    Account=9876
    AllocID=ZZZ
    OrderReference=10001
    OrderCapacity=Agency
```

Trade Capture Report Ack - Reject

```
TradeReportID=A12
ExecType=Rejected
TradeRptStatus=Rejected
TradeReportRejectReason=Invalid
Party Information
Text=Explanation...
```

Trade Capture Report Ack

```
TradeReportID=A12
SecondaryTradeReportID=013456A7
ExecType=Accepted
TradeRptStatus=Accepted
Account=9876
AllocID=ZZZ
```

9.5 Message Details

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

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Client-generated identifier
487	TradeReportTransType	Q	Valid values: 0 = New
856	TradeReportType	Q	Valid values: 0 = Submit
828	TrdType	Q	For valid values, please see 9.3.1.2
570	PreviouslyReported	Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
1116	NoRootPartyIDs		Number of party id entries (used for on-behalf-of transactions)
→	1117 RootPartyID	Q	Party identifier.
→	1118 RootPartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	1119 RootPartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 12 = Executing Trader
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
32	LastQty	Y	Traded quantity
31	LastPx	Y	Trade Price
21100	LastUnderlyingPx		
75	TradeDate	Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
60	TransactTime	Y	Time of execution/order creation
64	SettlDate		Settlement Date
552	NoSides	Y	Set to 1, only counterparty given
→	54 Side	Y	Valid values: 1 = Buy 2 = Sell
→	37 OrderID	Y	Required in FIX, but ignored
→	453 NoPartyIDs	Q	Normally set to 1 (counterparty). Can be set to 2 if trade is to be given up on entry.
→	→ 448 PartyID	Q	Counterparty ID/Take-up Firm

→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 14 = Giveup Clearing Firm 17 = Contra Firm 28 = Custodian 83 = Clearing Account
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace.
→	528	OrderCapacity			Account type. From Order. **NOTE:** OrderCapacity is not used by derivatives market
→	70	AllocID			Optional pass-thru field set by client and echoed back by marketplace.
→	483	TransBkdTime			Time of agreement.
→	20199	OpenCloseIndicator			The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer			Y	

9.5.2 Trade Capture Report – Two-Party Report (in)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Client-generated identifier
487	TradeReportTransType	Q	Valid values: 0 = New
856	TradeReportType	Q	Valid values: 0 = Submit
828	TrdType	Q	For valid values, please see 9.3.1.2.
570	PreviouslyReported	Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource must be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
32	LastQty	Y	Traded quantity
31	LastPx	Y	Trade Price
21100	LastUnderlyingPx		
75	TradeDate	Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
60	TransactTime	Y	Time of execution/order creation

64	SettlDate				Settlement Date
552	NoSides			Y	Set to 2 for two-party reports
→	54	Side		Y	Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX, but ignored
→	453	NoPartyIDs		Q	
→	→	448	PartyID	Q	Reporting party/Counterparty ID
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 12 = Executing Trader 14 = Giveup Clearing Firm 17 = Contra Firm 28 = Custodian 83 = Clearing Account
→	483	TransBkdTime			Time of agreement. **NOTE:** Can only set on the reporting party side.
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace. Only valid for PartyRole = Executing Firm or Contra Firm)
→	528	OrderCapacity			Account type. From Order. **NOTE:** OrderCapacity is not used by derivatives market
→	70	AllocID			Optional pass-thru field set by client and echoed back by marketplace. Only valid for PartyRole = Executing Firm or Contra Firm)
→	20199	OpenCloseIndicator			The open/close indicator controls if the resulting position after a trade should be an opening or a closing position. 0 = Default for the account (default value) 1 = Open 2 = Close Values between 128 and 255 are used to set the open/close indicator per leg for combination instruments. See section 6.6.10 for details.
	Standard Trailer			Y	

9.5.3 Trade Capture Report Ack (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
818	SecondaryTradeReportID		Genium INET order number.
150	ExecType	Y	Valid values: 0 = Accepted
939	TradeRptStatus	Q	Valid values: 0 = Accepted
55	Instrument/Symbol	Q	OMNet short name
48	Instrument/SecurityID	Q	Orderbook ID

22	Instrument/SecurityIDSource		Q	Valid values: M = Marketplace-assigned identifier
70	AllocID			Optional pass-thru field set by client and echoed back by marketplace.
1	Account			Optional pass-thru field set by client and echoed back by marketplace.
528	OrderCapacity			Account type, set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by derivatives market
1116	NoRootPartyIDs			Number of party id entries (used for on-behalf-of transactions and/or AFK/Custodian identifier.
→	1117	RootPartyID	Q	Party identifier.
→	1118	RootPartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	1119	RootPartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 14 = Giveup Clearing Firm 28 = Custodian 83 = Clearing Account
797	CopyMsgIndicator			Set to 'Y' on Drop Copy messages
	Standard Trailer		Y	

9.5.4 Trade Capture Report Ack – Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
150	ExecType	Y	Valid values: 8 = Rejected
939	TradeRptStatus	Q	Valid values: 1 = Rejected
751	TradeReportRejectReason	Q	Valid values: 1 = Invalid Party Information 2 = Unknown Instrument 3 = Unauthorized To Report Trades 4 = Invalid Trade Type 99 = Other
55	Instrument/Symbol	Y	**NOTE:** Set to [N/A]
58	Text		Can contain error message
	Standard Trailer	Y	

9.5.5 Trade Capture Report – Notification to Counterparty (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Identifier assigned by marketplace
487	TradeReportTransType	Q	Valid values: 0 = New
856	TradeReportType	Q	Valid values: 1 = Alleged
828	TrdType	Q	For valid values, please see 9.3.1.2.

573	MatchStatus			Q	Valid values: 1= Uncompared, unmatched or unaffirmed
570	PreviouslyReported			Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol			Q	OMNet short name
48	Instrument/SecurityID			Q	Orderbook ID
22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
21100	LastUnderlyingPx				
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	Time of execution/order creation
64	SettlDate				Settlement Date
552	NoSides			Y	Always 1 Side
→	54	Side		Y	Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	OrderID is required in FIX, but set to “NONE”
→	453	NoPartyIDs		Q	Always set to 1
→	→	448	PartyID	Q	Counterparty ID (reporting party)
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 7 = Entering Firm 17 = Contra Firm 36 = Entering Trader
→	483	TransBkdTime			Time of agreement.
797	CopyMsgIndicator				Set to ‘Y’ on Drop Copy messages
	Standard Trailer			Y	

9.5.6 Trade Capture Report – Delete Notification to Counterparty (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Identifier assigned by marketplace
572	TradeReportRefID		TradeReportID of previous notification to be modified or cancelled.
487	TradeReportTransType	Q	Valid values: 1 = Cancel
856	TradeReportType	Q	Valid values: 1 = Alleged
828	TrdType	Q	For valid values, please see 9.3.1.2.
573	MatchStatus	Q	Valid values: 1= Uncompared, unmatched or unaffirmed

570	PreviouslyReported			Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol			Q	OMNet short name
48	Instrument/SecurityID			Q	Orderbook ID
22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
21100	LastUnderlyingPx				
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	Time of execution/order creation
64	SettlDate				Settlement Date
552	NoSides			Y	Always 1 Side
→	54	Side		Y	Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	OrderID is required in FIX, but set to “NONE”
→	453	NoPartyIDs		Q	Always set to 1
→	→	448	PartyID	Q	Counterparty ID (reporting party)
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 7 = Entering Firm 17 = Contra Firm 36 = Entering Trader
→	483	TransBkdTime			Time of agreement.
797	CopyMsgIndicator				Set to ‘Y’ on Drop Copy messages
	Standard Trailer			Y	

9.5.7 Trade Capture Report – Deal on Hold Accepted (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Identifier assigned by marketplace
1003	TradeID	Q	Unique identifier for trade. **NOTE:** For OTC trades, this is formatted as a single hex-encoded 64-bit value.
572	TradeReportRefID	Q	From inbound TCR
818	SecondaryTradeReportID	Q	Genium INET order_number. Also present in previous TCR Ack message.
487	TradeReportTransType	Q	Valid values: 2 = Replace 4 = Reverse
856	TradeReportType	Q	Valid values: 0 = Submit

828	TrdType				
573	MatchStatus			Q	Valid values: 0 = Compared, matched or affirmed 1 = Uncompared, unmatched, or unaffirmed
880	TrdMatchID			Q	Match ID assigned by the matching engine.
570	PreviouslyReported			Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol			Q	OMNet short name
48	Instrument/SecurityID			Q	Orderbook ID
22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	**NOTE:** Contains Time of Trade Execution
552	NoSides			Y	Always 2 Sides
→	54	Side		Y	Side. Valid values:
					1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX. Set to “NONE”.
→	453	NoPartyIDs		Q	Number of party id entries
→	→	448	PartyID	Q	party identifier
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 = Giveup Clearing Firm (Takeup Firm) 17 = Contra Firm 36 = Entering Trader 38 = Position Account 83 = Clearing Account 1001 = Confirmed by Firm 1002 = Confirmed by User 1003 Reported by Firm 1004 Reported by User 1005 = Affirmed by Firm 1006 = Affirmed by User 1007 = Give-up Account
→	483		TransBkdTime		Time of agreement.
→	20009		OrderReference		Order Reference pass-thru field. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm

→	77	PositionEffect		Defines the *requested* position update for the account. Valid values:
				C = Close
				O = Open
				D = Default
				M = Mandatory Close

→	70	AllocID		Optional pass-thru field set by client and echoed back by marketplace. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
715	ClearingBusinessDate			
21102	DealNumber			Contains the numeric Genium INET deal_number
21103	TradeReportState		Q	Current state of the OTC Trade Report. Valid values: 0 = None 1 = Unmatched 2 = Paired 3 = Matched 4 = Cancelled 5 = Rejected 6 = Novated 7 = Terminated 8 = Deleted
21104	TradeReportSubState		Q	Current sub state of the OTC Trade Report. Valid values: 0 = None 1 = Pending Cancel 3 = Pending Termination 4 = Netted to zero 13 = Pending Clearing Member Acceptance 14 = Rejected by Clearing Member 15 = Pending Clearinghouse Confirmation 16 = Pending Clearinghouse Auto Confirm 17 = Pending Clearing 18 = Cancelled by Counterpart 19 = Ongoing Clearinghouse Check 20 = Auto Take Up Check Ongoing 21 = Auto Take Up Rejected

21105	TradeReportInstrType		Type of trade report. Valid values: 0 = None 1 = Standard 2 = Tailormade 3 = Fixed Income 4 = Discount Security 5 = FRA 6 = IR Swap 7 = FX 8 = Cash 9 = Repo 10 = Agreement 11 = SSI 12 = Equity 13 = XCUR Swap
21106	TraderReportReason		The reason a trade report is in a certain state or the action to a trade report. Valid values: 0 = None 1 = Counterparty has cancelled 2 = pending counterparty cancel 3 = counterparty has terminated 4 = pending counterparty termination 5 = Party Clearing Member 6 = Counterparty Clearing Member 7 = Party lacks collateral 8 = counterparty lacks collateral 9 = old account lacks collateral 10 = New account lacks collateral

			11 = Both accounts are lacking collateral 12 = manually confirmed by clearing house 13 = manually rejected by clearing house 14 = Automatic end of day cleanup 15 = rejected by counterparty 17 = exposure exceeded and lacking collateral 18 = record update 19 = Confirmation Due on Termination Date 20 = Configuration Error 21 = Party Exposure Limit Exceeded 22 = Counterparty Exceeded Exposure Limit 23 = Member defined exposure limit exceeded 24 = Old Account Exposure Limit Exceeded 25 = New Account Exposure Limit Exceeded 26 = Both Accounts Exposure Limit Exceeded 27 = Pending Compression 28 = Counterparty has updated
21107	AuthorizationState		Valid values: 0 = None 1 = Autorized 2 = Needed 3 = Not needed
21108	AffimationState		Valid values: 0 = Not required 1 = Holding 2 = Affirmed by Party 3 = Automatically Affirmed 4 = Rejected 5 = Auto Limit Exceeded
21109	DeliveryUnit		
21110	OrigClearingBusinessDate		
	Standard Trailer	Y	

9.5.8 Trade Capture Report Ack – Trade Cancel (out)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
487	TradeReportTransType	Q	Valid values: 1 = Cancel
818	SecondaryTradeReportID	Q	Genium INET order number.
150	ExecType	Y	Type of Execution being reported. Valid values: 4 = Canceled
939	TradeRptStatus	Q	Valid values: 0 = Accepted
55	Instrument/Symbol	Q	Short name of security

48	Instrument/SecurityID	Q	
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
70	AllocID		Optional pass-thru field set by client and echoed back by marketplace.
1	Account		Optional pass-thru field set by client and echoed back by marketplace.
528	OrderCapacity		Account type, set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by derivatives market
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
	Standard Trailer	Y	

9.5.9 Trade Capture Report – Trade cancel (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = AE
571	TradeReportID		Y	Client-generated identifier
572	TradeReportRefID			TradeReportID of report to cancel. Can be used instead of SecondaryTradeReportRefID.
881	SecondaryTradeReportRefID			Can be set to SecondaryTradeReportID (Genium INET order number) of the previously received Trade Capture Report Ack, This is the preferred identifier since it requires no lookups.
487	TradeReportTransType		Q	Valid values: 1 = Cancel
856	TradeReportType		Q	Valid values: 0 = Submit
570	PreviouslyReported		Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
1116	NoRootPartyIDs			Number of party id entries (used for on-behalf-of transactions)
→	1117	RootPartyID	Q	Party identifier.
→	1118	RootPartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	1119	RootPartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 12 = Executing Trader
55	Instrument/Symbol			OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48	Instrument/SecurityID			Orderbook ID
22	Instrument/SecurityIDSource			Valid values: M = Marketplace-assigned identifier
32	LastQty		Y	Not validated
31	LastPx		Y	Not validated

75	TradeDate			Y	Trade Date. Must be set to a valid date. Required in FIX but ignored
60	TransactTime			Y	Time of execution/order creation
552	NoSides			Y	Set to 1, only counterparty given
→	54	Side		Y	Counterparty Side. Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX, but ignored
→	453	NoPartyIDs		Q	Always set to 1 (counterparty)
→	→	448	PartyID	Q	Counterparty ID
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 17 = Contra Firm
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace. Only valid on the reporting Side (where PartyRole=Executing Firm)
→	70	AllocID			Optional pass-thru field set by client and echoed back by marketplace. Only valid on the reporting Side (where PartyRole=Executing Firm)
	Standard Trailer			Y	

9.5.10 Trade Capture Report Ack – Trade Cancel Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
487	TradeReportTransType	Q	Valid values: 1 = Cancel
150	ExecType	Y	Type of Execution being reported. Valid values: 8 = Rejected
939	TradeRptStatus	Q	Valid values: 1 = Rejected
751	TradeReportRejectReason	Q	Valid values: 99 = Other
55	Instrument/Symbol	Y	**NOTE:** Set to [N/A]
58	Text		Free text description of reject
	Standard Trailer	Y	

9.5.11 Trade Capture Report – Rectify Confirmed Trade (in)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Client-generated identifier
1003	TradeID	Q	NASDAQ OMX Extension: Unique identifier for trade. **NOTE:** The first part is ignored by the system,

						so a TradeID of 0:x, where x is the trade number, can be entered.
487	TradeReportTransType				Q	Valid values: 2 = Replace
856	TradeReportType				Q	Valid values: 0 = Submit
570	PreviouslyReported				Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol					OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48	Instrument/SecurityID					Orderbook ID
22	Instrument/SecurityIDSource					Valid values: M = Marketplace-assigned identifier
32	LastQty				Y	Not validated
31	LastPx				Y	Not validated
75	TradeDate				Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
60	TransactTime				Y	
552	NoSides				Y	Set to 1, only own side given
→	54	Side			Y	Own Side. Valid values: 1 = Buy 2 = Sell
→	37	OrderID			Y	Required in FIX, but ignored
→	826	TradeAllocIndicator				Identifies how the trade is to be allocated. Valid values: 6 = Trade Posting
→	78	NoAllocs			Q	Can be set to more than 1 if the trade is to be split into multiple accounts.
→	→	79	AllocAccount		Y	Account the trade should be posted to.
→	→	756	NoNested2PartyIDs		Q	Will always be set to 1.
→	→	→	757	Nested2PartyID		Clearing firm id (owner of account given in tag 79).
→	→	→	758	Nested2PartyIDSource		Valid values: D = Propr. Code
→	→	→	759	Nested2PartyRole		Valid values: 4 = Clearing Firm
→	→	467	IndividualAllocID			Pass-thru field set by client and echoed back by marketplace.
→	→	528	OrderCapacity		Y	Account type, set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by derivatives market
	Standard Trailer				Y	

9.5.12 Trade Capture Report Ack – Rectify Confirmed Trade (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
487	TradeReportTransType	Q	Valid values: 2 = Replace
1003	TradeID	Q	NASDAQ OMX Extension: Unique identifier for trade.
150	ExecType	Y	Type of Execution being reported. Valid values: G = Trade Correct
939	TradeRptStatus	Q	Valid values: 0 = Accepted
55	Instrument/Symbol	Q	Short name of security
48	Instrument/SecurityID	Q	
22	Instrument/SecurityIDSource	Q	Valid values: M = Marketplace-assigned identifier
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
	Standard Trailer	Y	

9.5.13 Trade Capture Report Ack – Rectify Confirmed Trade Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AR
571	TradeReportID	Y	The client-generated identifier
487	TradeReportTransType	Q	Valid values: 2 = Replace
150	ExecType	Y	Type of Execution being reported. Valid values: 8 = Rejected
939	TradeRptStatus	Q	Valid values: 1 = Rejected
751	TradeReportRejectReason	Q	Valid values: 99 = Other
55	Instrument/Symbol	Y	**NOTE:** Set to [N/A]
58	Text		Free text description of reject
	Standard Trailer	Y	

9.5.14 Allocation Instruction – Give up Request (in)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = J
70	AllocID		Y	Client-generated identifier
71	AllocTransType		Q	Valid values: 0 = New
626	AllocType		Q	Valid values: 17 = Give-Up
124	NoExecs		Q	1
	32	LastQty	Q	Required in FIX, ignored by the system.

	1003		TradeID	Q	NASDAQ OMX Extension: Unique identifier for trade
54			Side	Y	Required in FIX. Ignored
55			Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48			Instrument/SecurityID		Orderbook ID
22			Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
53			Quantity	Y	Required in FIX but ignored.
75			TradeDate	Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
78			NoAllocs	Y	Set to 1
→	79		AllocAccount	Y	Optional account (of the participant the trade is given up to). **NOTE:** This field is required in FIX. Must be set to "NONE" if no value is desired.
	80		AllocQty	Q	Quantity of the trade to be given up. This does not have to be the full trade quantity.
→	539		NoNestedPartyIDs	Q	Must be set to 1.
→	→	524	NestedPartyID	Q	ID of the participant the trade should be given up to.
→	→	525	NestedPartyIDSource	Q	Valid values: D = Propr. Code
→	→	538	NestedPartyRole	Q	Valid values: 14 = Giveup Clearing Firm (firm to which the trade is given up)
→	12		Commission		
→	13		CommType		Valid values: 3 = Absolute
58			Text		Contains user supplied text as information to the receiver. Max 30 chars.
			Standard Trailer	Y	

9.5.15 Allocation Report Ack – Reject (out)

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AT
755	AllocReportID	Y	Unique identifier for this message
70	AllocID	Y	The AllocID set by the requestor.
71	AllocTransType	Q	Valid values: 0 = New
87	AllocStatus	Q	Valid values: 1 = Block level reject
88	AllocRejCode	Q	Valid values: 99 = Other
58	Text		Free text description of reject

	Standard Trailer	Y	

9.5.16 Allocation Report – Give up Notification (out)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = AS
755	AllocReportID			Unique identifier for this message
70	AllocID			To the give up requestor this will be set to the AllocID sent in. Will not be set for other recipients.
793	SecondaryAllocID		Q	Give_up_number.
71	AllocTransType		Q	Valid values: 0 = New
794	AllocReportType		Q	Valid values: 15 = Give-Up 16 = Take-Up
87	AllocStatus		Y	Identifies the status off allocation. Valid values: 6 = allocation pending 7 = reversed 9 = claimed 10 = refused 14 = reversal pending
715	ClearingBusinessDate			
54	Side		Q	Valid values: 1 = Buy 2 = Sell
124	NoExecs		Q	1
→	32	LastQty	Y	Required, not used.
→	31	LastPx		Deal price
→	1003	TradeID	Q	NASDAQ OMX Extension: Unique identifier for trade
55	Instrument/Symbol			OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48	Instrument/SecurityID			Orderbook ID
22	Instrument/SecurityIDSource			Valid values: M = Marketplace-assigned identifier
53	Quantity		Y	Required in FIX but ignored.
30	LastMkt			Defines the trade venue
6	AvgPx		Y	Set to 0
453	NoPartyIDs		Q	Set to 1
→	448	PartyID	Q	Participant ID for firm giving up the trade
→	447	PartyIDSource	Q	Valid values: D = Proprietary
→	452	PartyRole	Q	Valid values: 1 = Executing Firm

75	TradeDate			Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
60	TransactTime				
78	NoAllocs			Y	Set to 1
→	79	AllocAccount			**NOTE:** This field is required in FIX. Will be set to “NONE” if no value exists.
→	80	AllocQty		Q	Quantity to be given up. Need not be full
→	539	NoNestedPartyIDs		Q	Will be set to 1
→	→	524	NestedPartyID	Y	Participant ID for the take up firm
→	→	525	NestedPartyIDSource	Q	Valid values: D = Proprietary
→	→	538	NestedPartyRole	Q	Valid values: 14 = Giveup Clearing Firm (firm to which the trade is given up)
→	12	Commission			
→	13	CommType			Valid values: 3 = Absolute
→	20014	ExternalTradeFeeType			The name of the trade fee type used to calculate the trade fee.
58	Text				Contains user supplied text as information to the receiver. Max 30 chars.
1040	SecondaryTradeID				NASDAQ OMX Extension: External trade number
1127	OrigSecondaryTradeID				NASDAQ OMX Extension: External trade number
855	SecondaryTrdType				NASDAQ OMX Extension: Contains Genium INET deal_source value.
	Standard Trailer			Y	

9.5.17 Allocation Instruction – Accept Give up Request (in)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = J
70	AllocID	Y	ID from Allocation Report – Give Up Notification message.
71	AllocTransType	Q	Valid values: 0 = New
626	AllocType	Q	Valid values: 18 = Take-Up
54	Side	Y	Required in FIX. Ignored
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
53	Quantity	Y	Required in FIX but ignored.

75	TradeDate			Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
78	NoAllocs			Y	Can be multiple if take up is to be distributed over several accounts
→	79	AllocAccount		Y	**NOTE:** This field is required in FIX. Must be set to “NONE” if no value is desired.
→	80	AllocQty		Q	
→	1047	AllocPositionEffect			
→	539	NoNestedPartyIDs		Q	Set to 1
→	→	524	NestedPartyID	Q	Identifier of the owner of the AllocAccount
→	→	525	NestedPartyIDSource	Q	Valid values: D = Proprietary
→	→	538	NestedPartyRole	Q	
→	161	AllocText			Contains customer_info
	Standard Trailer			Y	

9.5.18 Allocation Instruction – Reject Give up Request (in)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = J
70	AllocID	Y	ID from Allocation Report – Give Up Notification message.
71	AllocTransType	Q	Valid values: 0 = New
626	AllocType	Q	Valid values: 19 = Refuse Take-Up
54	Side	Y	Required in FIX. Ignored
55	Instrument/Symbol		OMNet short name. Either Symbol or SecurityID+SecurityIDSource needs to be set.
48	Instrument/SecurityID		Orderbook ID
22	Instrument/SecurityIDSource		Valid values: M = Marketplace-assigned identifier
53	Quantity	Y	Required in FIX but ignored.
75	TradeDate	Y	Trade Date. Must be set to a valid date. Required in FIX but ignored.
58	Text		Contains user supplied text as information to the receiver. Max 30 chars.
	Standard Trailer	Y	

10 Trade Confirmation and Management

A confirmed trade occurs when orders or quotes are executed and when privately negotiated trades are approved. The marketplace publishes confirmed trades to counterparties and possibly to other actors involved in the downstream processing of trades. Such actors can include:

- Broker back-office
- Broker clearing firms
- Clearing houses, Central Counter Parties (CCP)
- Central Securities Depositories (CSD)

Subject to marketplace rules, users are also allowed to request amendments and cancelation of previsouly confirmed trades.

NOTE: The Execution Report message is also used to report fills, but this message is primarily intended as responses to orders and quotes, i.e. for front-office use. The Trade Capture Report message as described in this chapter is primarily intended for actors that process trades in the downstream part of the transaction chain – and thereby designed to contain complete trade information. Market Data messages are used to publish public trade information for so called trade tickers.

NOTE 2: Due to the nature of Confirmed Trades, they are not normally sent on an order entry and trade reporting FIX session. But it is possible to enable the publication of Confirmed Trades on any FIX session.

10.1 Trade Confirmation features

10.1.1 Marketplace unsolicited modification of a confirmed trade

The marketplace may need to modify a trade after it has been confirmed. In this scenario, a client will receive two Trade Capture Report-confirmation messages. The first Trade Capture Report received will be a *reversal* of the original confirmation. It will have TradeReportTransType set to *Reverse*. The Side field will be the opposite of the original transaction (the logic is that the reversal should net out the original trade report).
The second Trade Capture Report *replaces* the original. It will have TradeReportTransType set to *Replace*.
The format of these two transactions follow the format of the trade it modifies (either auto-matched trade or confirmation). See sections 10.4.1 and 10.4.2 respectively for message details.

NOTE: All types of confirmed trades (including auto-matched trades) can be modified as described above.

10.1.2 Amending a Confirmed Trade

The marketplace may allow confirmed trades to be amended (also called rectified). The exact rules of how and when a confirmed trade is allowed to be changed are out of scope for this specification.
The following parameters are allowed to change:

- Account
- AllocID/CustomerInfo (pass-though information)

In addition, the trade can be split into multiple accounts using the rectify trade transaction.

The Trade Capture Report – Rectify Confirmed Trade message is used to rectify a trade (see section 9.5.11 for message details). The fields within the NoAllocs repeating group is used to change the trade parameters listed above. To split a trade into multiple accounts, use multiple instances of this repating group.

10.1.3 Give-Ups

It is possible to give up a confirmed trade to another member. Use the Allocation Instruction message to request give-up. The Take-Up gets notified via a Allocation Report message. The Take-Up firm uses the Allocation Instruction message to accept or reject the give-up.

10.1.4 Automatic Give-Ups

A trade can be automatically given up to another account. An automatic give up is executed if enabled by the CCP for the product and the take up account. Automatic give up can be initated from the Trade Capture Report message and the New Order Single

In Trade Capture Report there are a number of tags used to specify the take up account. For One-Party Report for Matching (in) the following tags are used:

- Tag 452 is set to 14
- Tag 448 state the take up member
- Tag 1119 is set to 83
- Tag 1117 state the take up clearing account ID
- Tag 20009 can be used for give up free text

For Two-Party Report (in):

- Tag 452 is set to 14
- Tag 448 state the take up member
- Tag 452 is set to 83
- Tag 448 state the take up clearing account ID

For New Order Single Specify the following:

- Tag 452 is set to 14
- Tag 448 state the take up member
- Tag 452 is set to 83
- Tag 448 state the take up clearing account ID

The automatic take up is notified to the take up party via an Allocation Report message. This message should be interpreted as informational message based on the text supplied in tag 58 in Allocation Instruction – Give Up Request (out).

10.1.5 Missing TargetSubIDs on some outbound Trade Capture Reports

Some outbound Trade Capture Report messages will not contain a TargetSubID commonly used to identify the trader that originally entered the transaction. The reason is that in some cases there has been no original transaction prior to receiving a TCR from the marketplace. The two situations are:

- When receiving a two-party confirmation to counterparty (in this case the counterparty reported the trade). See section 10.4.3 for message details.
- When receiving a notification to counterparty (in this case the counterparty has issued a one-party report and the counterparty receives a notification. See section 9.5.5 for message details.
- When receiving a Delete Notification to counterparty. See section 9.5.6 for message details.

10.1.6 Timestamps

10.1.6.1 Settlement Date

SettlDate (64) contains the Settlement date.

10.1.6.2 Time of Agreement

Time of agreement is shown by the TransBkdTime (483) field.

10.1.6.3 Time of Execution

Time of Execution is shown by the TransactTime (60) field.

10.1.6.4 Deferred Publication Time

The DeferredPublicationTime (20013) field contains the *number of minutes* the publication of this trade will be delayed. The time is relative to time of agreement (TransBkdTime, tag 483).
NOTE: A value of -1 means until end of day.

10.1.7 Aggressor Indicator

The AggressorIndicator (1057) field is set on auto-matched trades to show which side is the aggressive side. It is found in the TrdCapRptSideGrp on the “own” side.

10.2 Identifiers

10.2.1 Trade Report ID

The TradeReportID (571) is similar to the ClOrdID used for orders and executions. A unique Trade Report ID must be set on all reported trades (TCR) inbound to the marketplace.
The marketplace sets its own TradeReportIDs on outbound TCRs (like confirmed trades).

10.2.2 Trade Report Reference ID

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

10.2.3 Secondary Trade Report ID

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

identifier to use when canceling a previous Trade Capture Report since it requires no lookup in the gateway.
Secondary Trade Report ID is also set in confirmations.

10.2.4 Trade Match ID

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

10.2.5 Trade ID

TradeID is an identifier unique per day, assigned by the marketplace on confirmed trades. TradeID is formatted as a string containing two hex-encoded integers separated by a colon ":". The TradeID is unique per day within the system.

10.2.6 Original Trade ID

The OrigTradeID (1126) is a field that is used when the marketplace publishes updates to confirmed trades. As the name suggests, it is used to refer to the Trade ID of the original trade. It has the same format as TradeID.
Whenever the marketplace modifies a confirmed trade this sequence of messages is followed:

1. A Trade Capture Report (TCR) reversing the previous trade is issued.
2. A TCR replacing the original is sent out.

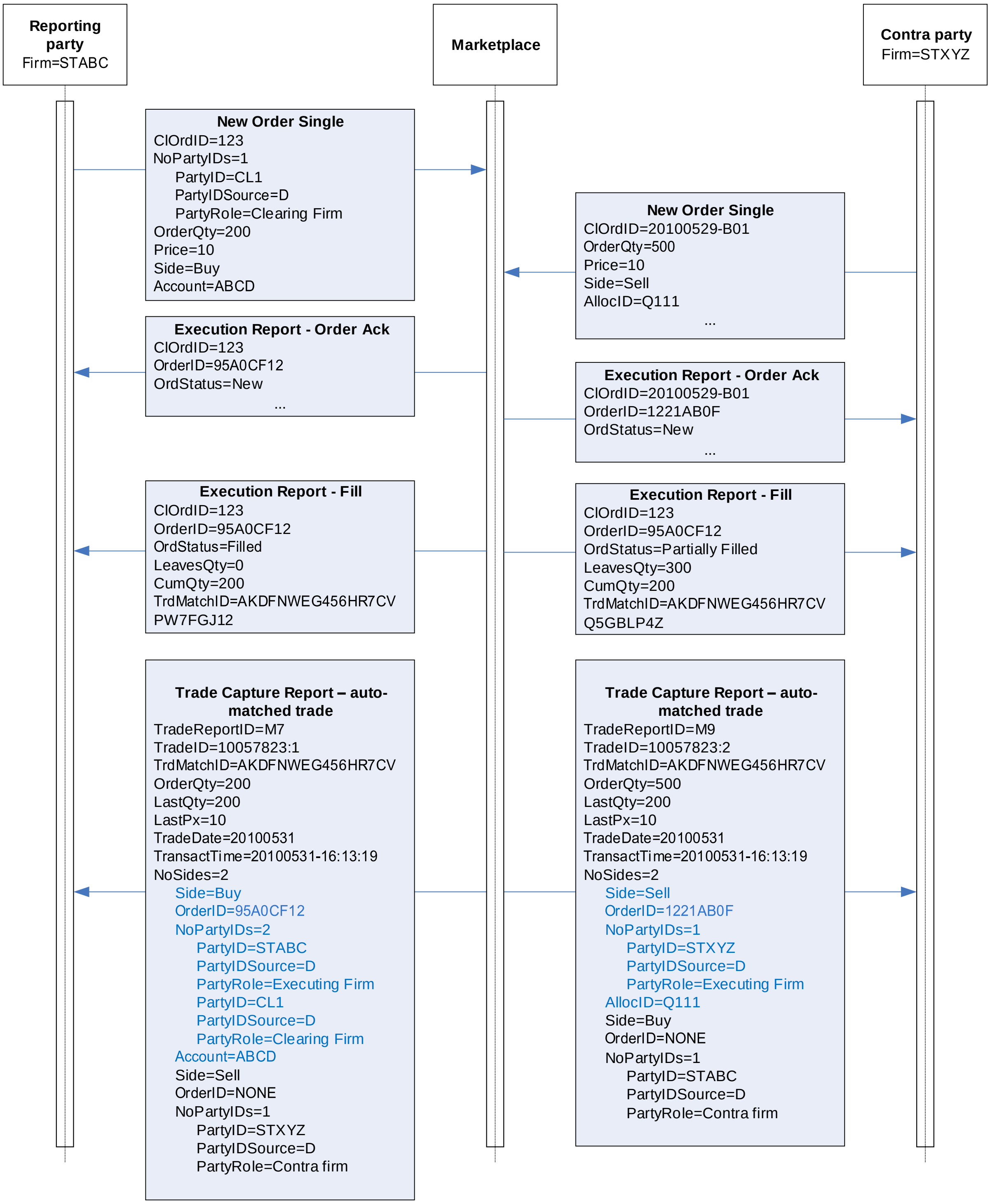
10.2.7 Deal ID

Deal ID (21000) contains the Genium INET deal_number.

10.3 Workflows

10.3.1 Trade Confirmation for an order that was matched

A regular order is placed in the book. When it is matched the client receives an Execution Report – Fill. In addition, at a later point a Trade Capture Report – auto-matched trade is received.
NOTE: in a typical setup, the confirmations are sent on a separate back-office FIX session to the client.
NOTE 2: For derivatives the Contra Side of the auto-matched trade will not be shown.



New Order Single
ClOrdID=123
NoPartyIDs=1
  PartyID=CL1
  PartyIDSource=D
  PartyRole=Clearing Firm
OrderQty=200
Price=10
Side=Buy
Account=ABCD

New Order Single
ClOrdID=20100529-B01
OrderQty=500
Price=10
Side=Sell
AllocID=Q111
...

Execution Report - Order Ack
ClOrdID=123
OrderID=95A0CF12
OrdStatus=New
...

Execution Report - Order Ack
ClOrdID=20100529-B01
OrderID=1221AB0F
OrdStatus=New
...

Execution Report - Fill
ClOrdID=123
OrderID=95A0CF12
OrdStatus=Filled
LeavesQty=0
CumQty=200
TrdMatchID=AKDFNWEG456HR7CVPW7FGJ12

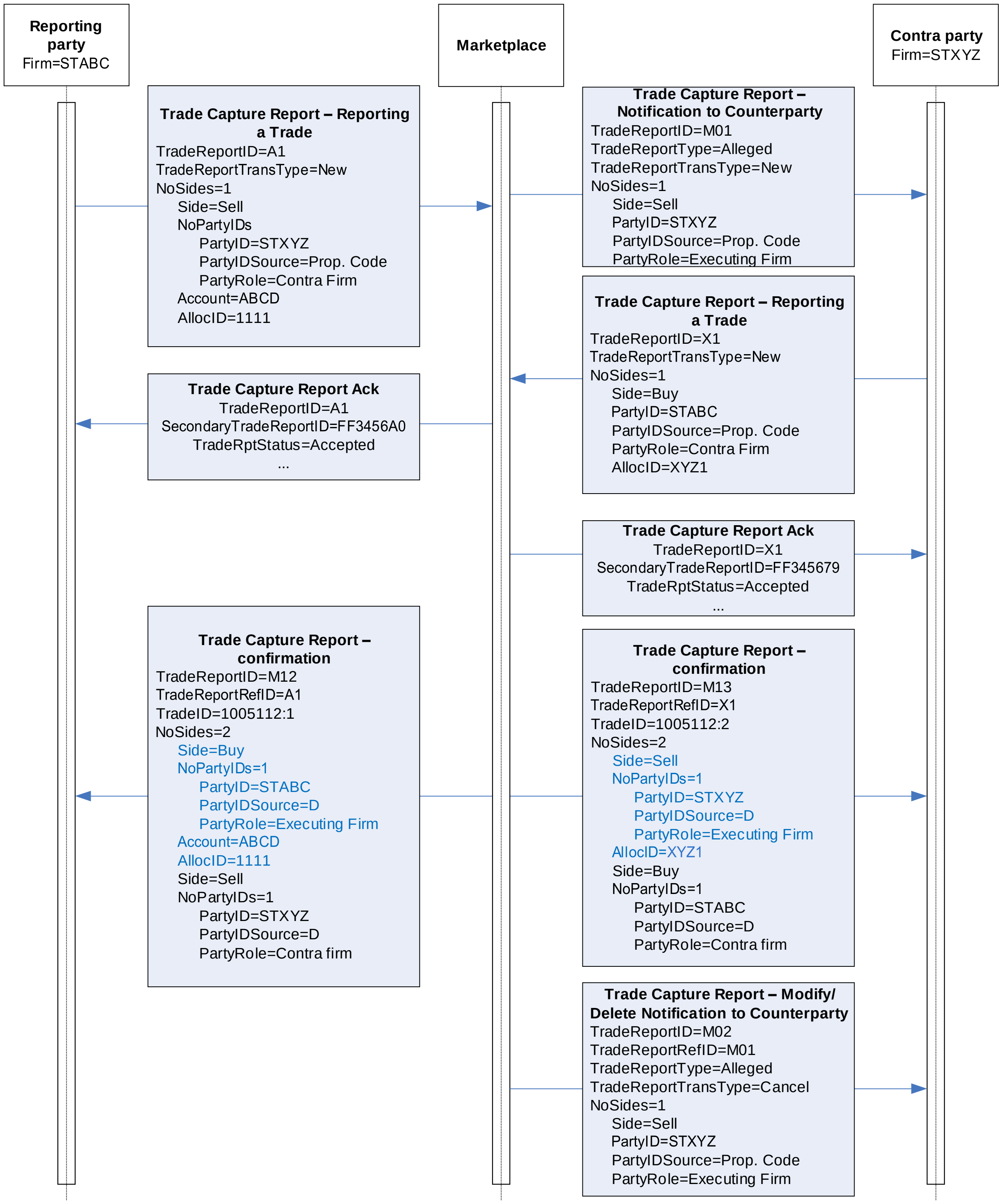
Execution Report - Fill
ClOrdID=123
OrderID=95A0CF12
OrdStatus=Partially Filled
LeavesQty=300
CumQty=200
TrdMatchID=AKDFNWEG456HR7CVQ5GBLP4Z

Trade Capture Report – auto-matched trade
TradeReportID=M7
TradeID=10057823:1
TrdMatchID=AKDFNWEG456HR7CV
OrderQty=200
LastQty=200
LastPx=10
TradeDate=20100531
TransactTime=20100531-16:13:19
NoSides=2
  Side=Buy
  OrderID=95A0CF12
  NoPartyIDs=2
    PartyID=STABC
    PartyIDSource=D
    PartyRole=Executing Firm
    PartyID=CL1
    PartyIDSource=D
    PartyRole=Clearing Firm
  Account=ABCD
  Side=Sell
  OrderID=NONE
  NoPartyIDs=1
    PartyID=STXYZ
    PartyIDSource=D
    PartyRole=Contra firm

Trade Capture Report – auto-matched trade
TradeReportID=M9
TradeID=10057823:2
TrdMatchID=AKDFNWEG456HR7CV
OrderQty=500
LastQty=200
LastPx=10
TradeDate=20100531
TransactTime=20100531-16:13:19
NoSides=2
  Side=Sell
  OrderID=1221AB0F
  NoPartyIDs=1
    PartyID=STXYZ
    PartyIDSource=D
    PartyRole=Executing Firm
  AllocID=Q111
  Side=Buy
  OrderID=NONE
  NoPartyIDs=1
    PartyID=STABC
    PartyIDSource=D
    PartyRole=Contra firm

10.3.2 Confirmation of a Reported Trade

Both parties have reported their side of the trade (Firm STABC is the *buyer*, STXYZ is the *seller*). The marketplace sends out a confirmation to both parties. When the first party sends in his report, the second party gets a notification. When the reports have matched, the notification gets cancelled.
NOTE: in a typical setup, the confirmations are sent on a separate back-office FIX session to the client.
NOTE 2: Notice how the inbound trade reports only contain the contra side. Firm STABC has to set his Account etc on the counterparty side (on the sell side even though he is the *buyer*). The outbound confirmations contain all these fields on the correct side (buy for firm STABC).
NOTE 3: Only fields relevant to the example are shown in the diagram.



Trade Capture Report – Reporting a Trade
TradeReportID=A1
TradeReportTransType=New
NoSides=1
Side=Sell
NoPartyIDs
PartyID=STXYZ
PartyIDSource=Prop. Code
PartyRole=Contra Firm
Account=ABCD
AllocID=1111

Trade Capture Report – Notification to Counterparty
TradeReportID=M01
TradeReportType=Alleged
TradeReportTransType=New
NoSides=1
Side=Sell
PartyID=STXYZ
PartyIDSource=Prop. Code
PartyRole=Executing Firm

Trade Capture Report – Reporting a Trade
TradeReportID=X1
TradeReportTransType=New
NoSides=1
Side=Buy
PartyID=STABC
PartyIDSource=Prop. Code
PartyRole=Contra Firm
AllocID=XYZ1

Trade Capture Report Ack
TradeReportID=A1
SecondaryTradeReportID=FF3456A0
TradeRptStatus=Accepted
...

Trade Capture Report Ack
TradeReportID=X1
SecondaryTradeReportID=FF345679
TradeRptStatus=Accepted
...

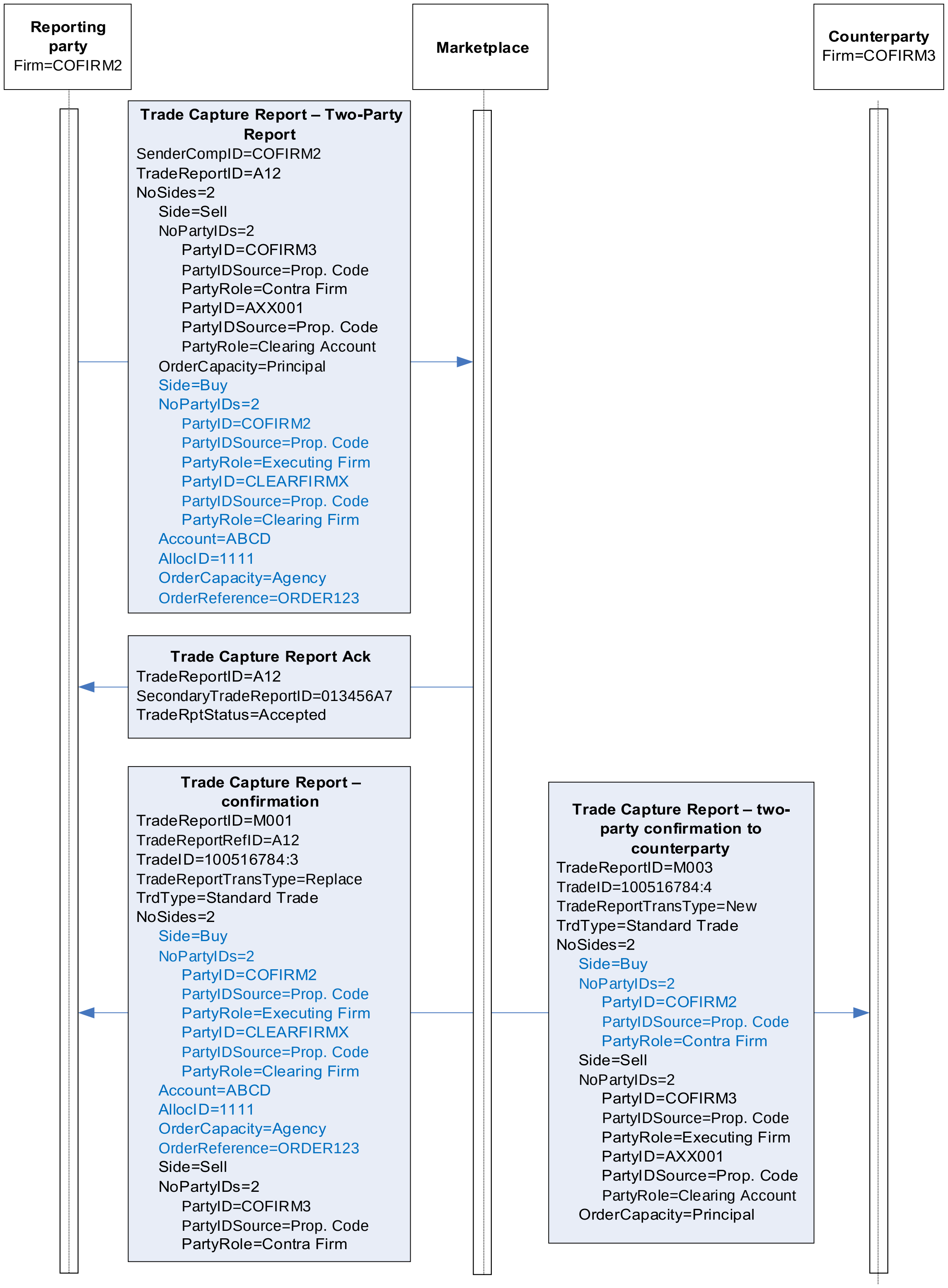
Trade Capture Report – confirmation
TradeReportID=M12
TradeReportRefID=A1
TradeID=1005112:1
NoSides=2
Side=Buy
NoPartyIDs=1
PartyID=STABC
PartyIDSource=D
PartyRole=Executing Firm
Account=ABCD
AllocID=1111
Side=Sell
NoPartyIDs=1
PartyID=STXYZ
PartyIDSource=D
PartyRole=Contra firm

Trade Capture Report – confirmation
TradeReportID=M13
TradeReportRefID=X1
TradeID=1005112:2
NoSides=2
Side=Sell
NoPartyIDs=1
PartyID=STXYZ
PartyIDSource=D
PartyRole=Executing Firm
AllocID=XYZ1
Side=Buy
NoPartyIDs=1
PartyID=STABC
PartyIDSource=D
PartyRole=Contra firm

Trade Capture Report – Modify/ Delete Notification to Counterparty
TradeReportID=M02
TradeReportRefID=M01
TradeReportType=Alleged
TradeReportTransType=Cancel
NoSides=1
Side=Sell
PartyID=STXYZ
PartyIDSource=Prop. Code
PartyRole=Executing Firm

10.3.3 Confirmation of a two-party Trade Report

The Entering party, COFIRM2, enters a two-party trade report. The counterparty is COFIRM3.
NOTE: If instead COFIRM1 enters the trade on behalf of COFIRM2, the SenderCompID is changed to COFIRM1. All other fields remain the same.
NOTE 2: Only fields relevant to the example are shown in the diagram.



Trade Capture Report – Two-Party Report

```
SenderCompID=COFIRM2
TradeReportID=A12
NoSides=2
    Side=Sell
    NoPartyIDs=2
        PartyID=COFIRM3
        PartyIDSource=Prop. Code
        PartyRole=Contra Firm
        PartyID=AXX001
        PartyIDSource=Prop. Code
        PartyRole=Clearing Account
    OrderCapacity=Principal
    Side=Buy
    NoPartyIDs=2
        PartyID=COFIRM2
        PartyIDSource=Prop. Code
        PartyRole=Executing Firm
        PartyID=CLEARFIRMX
        PartyIDSource=Prop. Code
        PartyRole=Clearing Firm
    Account=ABCD
    AllocID=1111
    OrderCapacity=Agency
    OrderReference=ORDER123
```

Trade Capture Report Ack

```
TradeReportID=A12
SecondaryTradeReportID=013456A7
TradeRptStatus=Accepted
```

Trade Capture Report – confirmation

```
TradeReportID=M001
TradeReportRefID=A12
TradeID=100516784:3
TradeReportTransType=Replace
TrdType=Standard Trade
NoSides=2
    Side=Buy
    NoPartyIDs=2
        PartyID=COFIRM2
        PartyIDSource=Prop. Code
        PartyRole=Executing Firm
        PartyID=CLEARFIRMX
        PartyIDSource=Prop. Code
        PartyRole=Clearing Firm
    Account=ABCD
    AllocID=1111
    OrderCapacity=Agency
    OrderReference=ORDER123
    Side=Sell
    NoPartyIDs=2
        PartyID=COFIRM3
        PartyIDSource=Prop. Code
        PartyRole=Contra Firm
```

Trade Capture Report – two-party confirmation to counterparty

```
TradeReportID=M003
TradeID=100516784:4
TradeReportTransType=New
TrdType=Standard Trade
NoSides=2
    Side=Buy
    NoPartyIDs=2
        PartyID=COFIRM2
        PartyIDSource=Prop. Code
        PartyRole=Contra Firm
    Side=Sell
    NoPartyIDs=2
        PartyID=COFIRM3
        PartyIDSource=Prop. Code
        PartyRole=Executing Firm
        PartyID=AXX001
        PartyIDSource=Prop. Code
        PartyRole=Clearing Account
    OrderCapacity=Principal
```

10.4 Message Details

10.4.1 Trade Capture Report – auto-matched trade (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name	Req’d	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Identifier assigned by marketplace
1003	TradeID	Q	TradeID is a unique identifier for a trade.
1040	SecondaryTradeID		Trade id assigned by external system.
1126	OrigTradeID		Used to refer to original trade in case of modifications
1127	OrigSecondaryTradeID		Original trade id assigned by external system.
487	TradeReportTransType	Q	Valid values: 0 = New 2 = Replace 4 = Reverse
856	TradeReportType	Q	Valid values: 0 = Submit
829	TrdSubType	Q	Further qualification of the trade type (omnet trade_type_c).
573	MatchStatus	Q	Valid values: 0 = Compared, matched or affirmed
730	SettlementPrice		Calculated price for settlement.
20017	WithHoldingTax		If repo trade is subject to witholding tax value of tax amount.
20018	RepoInterestAmt		Repo Trades interest amount.
1703	NoCollateralAmounts		Number of collateral amount entries.
→	1704 CurrentCollateralAmount		Currency denomination of value
20007	CorrespondingPrice		*NASDAQ Extension:* Corresponding Price/Yield for fixed income related trades.
20008	Consideration		*NASDAQ Extension:* Consideration/Settlement Amount for fixed income related trades.
880	TrdMatchID	Q	Match ID assigned by the matching engine.
20034	ComboMatchID		Combination Match ID assigned by the matching engine.
570	PreviouslyReported	Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol	Q	OMNet short name
48	Instrument/SecurityID	Q	Orderbook ID

22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
38	OrderQtyData/OrderQty				Will not be set for trades originating from quotes
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
21100	LastUnderlyingPx				
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	**NOTE:** Contains Time of Trade Execution
552	NoSides			Y	Either 1 (own side only) or 2 (both sides)
→	54	Side		Y	Side. Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	OrderID on own Side. Set to “NONE” on counterparty side.
→	453	NoPartyIDs		Q	Number of party id entries
→	→	448	PartyID	Q	party identifier
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 = Giveup Clearing Firm (Takeup Firm) 17 = Contra Firm 28 = Custodian 36 = Entering Trader
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace. Only set on the own Side (where PartyRole=Executing Firm). **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
→	528	OrderCapacity			Account type. Pass-thru field set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by derivatives market
→	70	AllocID			Optional pass-thru field set by client and echoed back by marketplace. Only set on the own Side (where PartyRole=Executing Firm). **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)

→	1057	AggressorIndicator		Indicates who is the aggressive party in the trade. Valid values: Y = Party is the aggressor N = Party is passive **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
381	GrossTradeAmt			Calculated trade value.
20015	OptionPremiumAmt			Premium value
715	ClearingBusinessDate			
855	SecondaryTrdType			Contains Genium INET deal_source value.
793	SecondaryAllocID			Contains Genium INET Give_up_number.
797	CopyMsgIndicator			Set to 'Y' on Drop Copy messages
	Standard Trailer		Y	

10.4.2 Trade Capture Report – confirmation (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = AE
571	TradeReportID		Y	Identifier assigned by marketplace
1003	TradeID		Q	TradeID is a unique identifier for a trade.
1040	SecondaryTradeID			Trade id assigned by external system.
1126	OrigTradeID			Used to refer to original trade in case of modifications
1127	OrigSecondaryTradeID			Original trade id assigned by external system.
572	TradeReportRefID		Q	From inbound TCR
818	SecondaryTradeReportID		Q	Genium INET order_number. Also present in previous TCR Ack message.
487	TradeReportTransType		Q	Valid values:2 = Replace 4 = Reverse
856	TradeReportType		Q	Valid values: 0 = Submit
828	TrdType			For valid values, please see 9.3.1.2.
829	TrdSubType		Q	Further qualification of the trade type (omnet trade_type_c).
573	MatchStatus		Q	Valid values: 0 = Compared, matched or affirmed
730	SettlementPrice			Calculated price for settlement.
20017	WithHoldingTax			If repo trade is subject to witholding tax value of tax amount.
20018	RepoInterestAmt			Repo Trades interest amount.
1703	NoCollateralAmounts			Number of collateral amount entries.
→	1704	CurrentCollateralAmount		Currency denomination of value
20007	CorrespondingPrice			*NASDAQ Extension:* Corresponding Price/Yield for fixed income related trades.

20008	Consideration				*NASDAQ Extension:* Consideration/Settlement Amount for fixed income related trades.
880	TrdMatchID			Q	Match ID assigned by the matching engine.
570	PreviouslyReported			Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol			Q	OMNet short name
48	Instrument/SecurityID			Q	Orderbook ID
22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
21100	LastUnderlyingPx				
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	**NOTE:** Contains Time of Trade Execution
64	SettlDate				Settlement date
552	NoSides			Y	Always 2 Sides
→	54	Side		Y	Side. Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX. Set to “NONE”.
→	453	NoPartyIDs		Q	Number of party id entries
→	→	448	PartyID	Q	party identifier
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 14 = Giveup Cleaaring Firm (Takeup Firm) 17 = Contra Firm 28 = Custodian 36 = Entering Trader
→	483	TransBkdTime			Time of agreement. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace. Only set on the own Side (where PartyRole=Executing Firm) **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
→	528	OrderCpacity			Account type. Pass-thru field set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by derivatives market

→	70	AllocID		Optional pass-thru field set by client and echoed back by marketplace. Only set on the own Side (where PartyRole=Executing Firm) **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
381	GrossTradeAmt			Calculated trade value.
20015	OptionPremiumAmt			Premium value
715				
855	SecondaryTrdType			Contains Genium INET deal_source value.
793	SecondaryAllocID			Contains Genium INET Give_up_number.
797	CopyMsgIndicator			Set to ‘Y’ on Drop Copy messages
	Standard Trailer		Y	

10.4.3 Trade Capture Report – two-party confirmation to counterparty (out) (Currently only supported for drop copy sessions)

Tag	FIX tag name		Req’d	Comment
	Standard Header		Y	MsgType = AE
571	TradeReportID		Y	Identifier assigned by marketplace
1003	TradeID		Q	TradeID is a unique identifier for a trade.
1126	OrigTradeID			Used to refer to original trade in case of modifications
1127	OrigSecondaryTradeID			Original trade id assigned by external system.
818	SecondaryTradeReportID		Q	Genium INET order_number.
487	TradeReportTransType		Q	Valid values: 0 = New 2 = Replace 4 = Reverse
856	TradeReportType		Q	Valid values: 0 = Submit
828	TrdType			For valid values, please see 9.3.1.2.
829	TrdSubType		Q	Further qualification of the trade type (omnet trade_type_c).
573	MatchStatus		Q	Valid values: 0 = Compared, matched or affirmed
730	SettlementPrice			Calculated price for settlement.
20017	WithHoldingTax			If repo trade is subject to witholding tax value of tax amount.
20018	RepoInterestAmt			Repo Trades interest amount.
1703	NoCollateralAmounts			Number of collateral amount entries.
→	1704	CurrentCollateralAmount		Currency denomination of value
20007	CorrespondingPrice			*NASDAQ Extension:* Corresponding Price/Yield for fixed

					income related trades.
20008	Consideration				*NASDAQ Extension:* Consideration/Settlement Amount for fixed income related trades.
880	TrdMatchID			Q	Match ID assigned by the matching engine.
570	PreviouslyReported			Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Instrument/Symbol			Q	OMNet short name
48	Instrument/SecurityID			Q	Orderbook ID
22	Instrument/SecurityIDSource			Q	Valid values: M = Marketplace-assigned identifier
32	LastQty			Y	Traded quantity
31	LastPx			Y	Trade Price
21100	LastUnderlyingPx				
75	TradeDate			Y	Always set to date of trade.
60	TransactTime			Y	**NOTE:** Contains Time of Trade Execution
64	SettlDate				Settlement date
552	NoSides			Y	Always 2 Sides
→	54	Side		Y	Side. Valid values: 1 = Buy 2 = Sell
→	37	OrderID		Y	Required in FIX. Set to “NONE”.
→	453	NoPartyIDs		Q	Number of party id entries
→	→	448	PartyID	Q	party identifier
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 7 = Entering Firm 14 = Giveup Clearing Firm (Takeup Firm) 28 = Custodian 36 = Entering Trader
→	483	TransBkdTime			Time of agreement. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
→	1	Account			Optional pass-thru field set by client and echoed back by marketplace. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
→	528	OrderCapacity			Account type. Pass-thru field set by client and echoed back by marketplace. **NOTE:** OrderCapacity is not used by

				derivatives market
→	70	AllocID		Optional pass-thru field set by client and echoed back by marketplace. **NOTE:** Only set on the own Side (where PartyRole=Executing Firm)
715	ClearingBusinessDate			
855	SecondaryTrdType			Contains Genium INET deal_source value.
793	SecondaryAllocID			Contains Genium INET Give_up_number.
797	CopyMsgIndicator			Set to 'Y' on Drop Copy messages
	Standard Trailer		Y	

11 Appendix A, NASDAQ OMX Extensions

This chapter details how this solution deviates from standard FIX 5.0 SP2. While great care has been taken to conform to the standard, a number of deviations are unavoidable to support all mechanisms provided by the host.

There are different types of deviations from the standard:

- Fields added. A few user defined fields had to be added to accommodate back-end functionality not present in FIX 5.0 SP2.
- Enumerated values added. Some fields have added enums.
- Field definition changed.

11.1 Added Fields

Field	Name	Added To Message	Comment
20034	ComboMatchID	Execution Report	Match ID
21060	Fineness	Security Definition Request	Price
20199	OpenCloseIndicator	New Order Single Order Cancel Replace Request Execution Report Trade Capture Report	int
20200	LegTrdMatchID	Execution Report	Match ID
20015	OptionPremiumAmt	Execution Report Trade Capture Report	Amt
21100	LastUnderlyingPx	Execution Report Trade Capture Report	Price
21101	UnderlyingColPx	New Order Single Order Cancel Replace Request Execution Report	Price
21102	DealNumber	Trade Capture Report	int
20009	OrderReference	Trade Capture Report	String
20014	ExternalTradeFeeType	Allocation Report	String
20002	DaysToPwdExpiry	Logon	int
21103	TradeReportState	Trade Capture Report	int
21104	TradeReportSubState	Trade Capture Report	int
21105	TradeReportInstrType	Trade Capture Report	Int
21106	TradeReportReason	Trade Capture Report	Int
21107	AuthorizationState	Trade Capture Report	Int
21108	AffirmationState	Trade Capture Report	Int
21109	DeliveryUnit	Trade Capture Report	int
21110	OrigClearingBusinessDate	Trade Capture Report	LocalMktDate
21113	DeferredPublicationTime	Security Definition	int

11.2 Added Enumerations

Enumeration	Added to Field Name	Comment
100 = Invalid BodyLength, session suspended 101 = HeartBt interval too low	SessionStatus	

11.3 Field Definition Changed

No field definitions changed.

12 Appendix B, Field length limitations

The following fields have a max length limit:

Tag Num	Field Name	max length	Comment
11	ClOrdID	20	
41	OrigClOrdID	20	
117	QuoteID	20	
320	SecurityRequestID	20	
70	AllocID	15	
1	Account	15	
79	AllocAccount	10	
448	PartyID	7	When PartyRole=Executing Firm or Contra Firm
448	PartyID	3	When PartyRole=Custodian
524	NestedPartyID	7	
757	Nested2PartyID	7	
571	TradeReportID	20	
572	TradeReportRefID	20	
881	SecondaryTradeReportRefID	20	
923	UserRequestID	20	
925	NewPassword	32	

13 Appendix C, Trade types

The following table contains the definitions of all values the TrdType (828) field can contain.

Value	Description
501	EQWS - PAY PIYASASI TOPTAN SATISLAR
514	ELT0 - SON ISLEM FIYATI
515	EQNS - PAY PIYASASI TS TAKAS DISARIDA
516	ELT6 - SON ISLEM FIYATI +/- %6
621	DIXF - BIST30 VIS OZEL EMIR
622	DIXO - BIST30 OPSIYON OZEL EMIR
623	DIXM - MINI BIST30 OPSIYON OZEL EMIR
624	DEF1 - PAY VIS OZEL EMIR
625	DEF2 - PAY VIS OZEL EMIR
626	DEO1 - PAY OPSIYON OZEL EMIR
627	DEO2 - PAY OPSIYON OZEL EMIR
628	DUSF - USDTRY VIS OZEL EMIR
629	DEUR - EURTRY VIS OZEL EMIR
630	DEUD - EURUSD VIS OZEL EMIR
631	DUSO - USDTRY OPSIYON OZEL EMIR
632	DXTR - GRAM ALTIN VIS OZEL EMIR
633	DXUS - ONS ALTIN VIS OZEL EMIR
634	DCOT - FIZIKI TES. PAMUK VIS OZEL EMIR
635	DWHT - AN. KIR. BUGDAY VIS OZEL EMIR
636	DNUL - OZEL EMIR GIRISI YOK
637	DEL - ELEKTRIK VIS OZEL EMIR
638	DSAS - SASX10 VIS OZEL EMIR
639	DHMS - HURDA CELIK VIS OZEL EMIR
640	DETF - FBIST ETF VIS OZEL EMIR
641	DIXR - BIST30 VIS OZEL EMIR (ROLL)
642	DUSR - USDTRY VIS OZEL EMIR (ROLL)
643	DONF - REPO VIS OZEL EMIR
644	DRBU - USDRUB VIS OZEL EMIR
645	DRUB - RUBTRY VIS OZEL EMIR
646	DCNU - USDCNH VIS OZEL EMIR
647	DCNH - CNHTRY VIS OZEL EMIR
648	DCYU - USDCNY VIS OZEL EMIR
649	DCNY - CNYTRY VIS OZEL EMIR
680	DMNS - MNSE10 VIS OZEL EMIR
681	DWHM - DURUM BUGDAY VIS OZEL EMIR
682	DXAP - FIZIKI TES. ALTIN VIS OZEL EMIR
683	DEUO - EURTRY OPSIYON OZEL EMIR
684	DELC - ELEKTRIK CASC VIS OZEL EMIR
685	DGUD – GBPUSD VIS OZEL EMIR

686	DXBF – BIST BANKA VIS OZEL EMIR
687	DXSF – BIST SINAI VIS OZEL EMIR
721	DA01 - BIST30 VIS ILAN PAZARI
722	DA02 - BIST30 OPSIYON ILAN PAZARI
723	DA03 - MINI BIST30 OPSIYON ILAN PAZARI
724	DA04 - PAY VIS ILAN PAZARI
725	DA05 - PAY VIS ILAN PAZARI
726	DA06 - PAY OPSIYON ILAN PAZARI
727	DA07 - PAY OPSIYON ILAN PAZARI
728	DA08 - USDTRY VIS ILAN PAZARI
729	DA09 - EURTRY VIS ILAN PAZARI
730	DA10 - EURUSD VIS ILAN PAZARI
731	DA11 - USDTRY OPSIYON ILAN PAZARI
732	DA12 - GRAM ALTIN VIS ILAN PAZARI
733	DA13 - ONS ALTIN VIS ILAN PAZARI
734	DA14 - FIZIKI TES.PAMUK VIS ILAN PAZARI
735	DA15 - AN. KIR. BUGDAY VIS ILAN PAZARI
737	DA16 - ELEKTRIK VIS ILAN PAZARI
738	DA17 - SASX10 VIS ILAN PAZARI
739	DA18 - HURDA CELIK VIS ILAN PAZARI
740	DA19 - FBIST ETF VIS ILAN PAZARI
743	DA20 - REPO VIS ILAN PAZARI
744	DA21 - USDRUB VIS ILAN PAZARI
745	DA22 - RUBTRY VIS ILAN PAZARI
746	DA23 - USDCNH VIS ILAN PAZARI
747	DA24 - CNHTRY VIS ILAN PAZARI
748	DA25 - USDCNY VIS ILAN PAZARI
749	DA26 - CNYTRY VIS ILAN PAZARI
750	DA27 - MNSE10 VIS ILAN PAZARI
751	DA28 - DURUM BUGDAY VIS ILAN PAZARI
752	DA29 - FIZIKI TES ALTIN VIS ILAN PAZARI
753	DA30 - EURTRY OPSIYON ILAN PAZARI
754	DA31 - ELEKTRIK CASC VIS ILAN PAZARI
812	FTR1 - BAP TRADE REPORT – CLEARED
818	FTR2 - BAP TRADE REPORT - NON-CLEARED
821	FTRC - BAP TR - CLEARED - 500k
831	FC1K - BAP TR - CLEARED - 1k
832	FC10 - BAP TR - CLEARED - 10k
833	FC50 - BAP TR - CLEARED - 50k
834	FCHK - BAP TR - CLEARED - 100k
841	FN1K - BAP TR - NON-CLEARED - 1k
842	FN10 - BAP TR - NON-CLEARED - 10k
843	FN50 - BAP TR - NON-CLEARED - 50k

844	FNHK - BAP TR - NON-CLEARED - 100k
901	KSNI – PMD (KMP) TR CLEARED IN (On Exchange)
902	KSNO – PMD (KMP) TR CLEARED OUT (Off Exch.)

14 Appendix D, Secondary Trade Types

Value	Description
0	Internal use. Trades reported directly to the clearing subsystem.
1	Matched by system, automatically.
2	Matched by system, manually.
3	Matched Outside Exchange, Different participants
4	Matched outside exchange, different participants, reg. by exchange.
5	Matched Outside Exchange, One participant
6	Matched outside exchange, one participant, reg. by exchange.
7	Combination order matched against another combination order when matched by the Exchange, electronically.
8	Deal in a Swap Box instrument.
9	Matched electronically, member internal.
10	Deal in a Swap Box instrument, member internal.
11	After market closure, outside system, different brokers
12	After market closure, outside system, different brokers, registered by the exchange.
13	After market closure, outside system, one broker
14	After market closure, outside system, one broker, registered by the exchange.
15	Internally created basis trade.
16	Reversing deal made by the exchange manually.
17	Basis trade.
18	Correction of trade.
19	Internally created.
20	Deal made at the end of an auction.
21	Private request for quote.
22	Package private request for quote.
23	Internally from combo.
24	Internally from TM.
25	Internally from average.
26	Internally from strip.
27	Delta hedge.
28	CL bundle deal.
32	Trade from Bulletin Board.
33	Trade from Bulletin Board, standard combo.
34	Trade from Bulletin Board, non-standard combo.
35	Trade from Bulletin Board, non-standard combo.

36	Tailor-made combination.
37	Non-standard combination.
38	Outside the Exchange, block trade facility.
39	Matched outside the Exchange, combinations.
40	Outside the Exchange, block trade facility.
41	No Deal Price.
42	Priority crossing.
43	Combination matched outright legs.
44	Matched outside exchange, broker.
53	Negotiated deal advertising board.
100	deal_source_imp_rotation
101	deal_source_imp_normal
102	deal_source_imp_out_of_sequence
103	deal_source_imp_cab_trade
104	deal_source_imp_combo_single
105	deal_source_imp_combo_mix
110	deal_source_fac_orig_order
111	deal_source_fac_counter_order
112	deal_source_exp_orig_order
113	deal_source_exp_counter_order
114	deal_source_unsolicited_order
115	deal_source_solicited_order
116	deal_source_block_order
117	deal_source_trade_rep
118	deal_source_trade_rep_no_settl
122	deal_source_imp_combo_buy_write
128	Trade resulting from an Aver- age Price Trade transaction.
129	Intermediate trade created in an Average Price Trade transaction.
131	Trade transfer.
132	Misclear.
133	Exchange for physical (EFP).
134	Spread trade.
135	Average price system (APS).
136	Adjustment without price.
137	Adjustment with price.
138	Deal executed at CTrade.
139	Cross product netting.
140	Committed transaction return.

Revision History

Date	Revision	Change Description
2014-02-21	0.1	Initial version.
2014-06-13	0.2	First revision for BIST Beta2.
2014-09-03	0.3	Changed the phrasing regarding dynamic trade report types and made BIST account fields reuired
2014-10-16	0.4	Added Giveup Clearing firm and Clearing Account to new order single, Execution Reports and Trade Capture Reports. Removed Trade Cancel And Trade Cancel Reject messages. Added post-trade messages: Allocation Instruction – Reject Give up Request Allocation Instruction – Accept Give up Request Allocation Report – Give up Notification Allocation Report Ack – Reject Allocation Instruction – Give up Request Trade Capture Report Ack – Rectify Confirmed Trade Reject Trade Capture Report Ack – Rectify Confirmed Trade Trade Capture Report – Rectify Confirmed Trade (in). Also added text regarding give up functionality and clearing accounts.
2014-11-18	0.5	Removed TMC and REPO registration information. Added new logo and Market Maker Supervision information. Removed Information about Trigger on session Orders and Open/Close Indicator
2014-12-05	0.6	Removed QuoteID to clordID in TCR mapping.
2015-01-26	0.7	Added tag 1094 to NoS and Execution Report.. Added at crossing as a valid TiF. Added a short description on how to enter Imbalance orders. Added TCR Trade Cancel (in) and TCR Trade Cancel reject messages.
2015-02-24	0.8	Added GTS related tags to NoS and added some more info regarding Market to Limit orders in auction phase and Long order Expiration.
2015-03-27	0.9	Changed encoding from Latin-1 to Latin-9
2015-03-30	0.9.1	Changed tags for quantity and quoteReqID in QuoteRequest-OUT.
2015-06-04	0.9.2	Added Order Capacity to trade cancel OUT message. Added description for custodian role, account and orderCapacity are no longer required for orderCancelReplace.
2015-07-10	0.9.3	Minor updates.
	0.9.4	Give up functionality is currently not supported.

Date	Revision	Change Description
2015-09-21	0.9.5	TCR Out (msgType = AE) is currently only supported on drop sessions. User Request messages are currently not supported. Account, OrderCapacity, PartyID (partygroup) are no longer allowed to be modified. Added at crossing to Order Expiry table. Added Appendix B, trade
		types table. Added Broker opt as a possible value for tag 378.
2015-10-05	0.9.6	Removed the drop copy section since we have a separate drop copy specification.
2015-10-28	0.9.7	Added PartyID to the list of fields that should be left out from the cancelReplace message if the field should not be changed.
2015-04-16	1.0.1	Initial version for phase 2
2015-09-14	1.0.2	Phase 2 specific changes and new changes done in phase 1 since the branch of documentation.
2016-02-08	1.0.3	Adding position effect, TRL value conversion, advertising orders and updating supported messages. Added description for how to enter trigger on session and on close orders.
2016-02-25	1.0.4	Added Comment to UnderlyingColPx and explanation for Trigger Instruction.
2016-03-10	1.0.5	Changed value for tag 59 and fixed typo in Chapter On Close Orders.
2016-04-05	1.0.6	Updated note for chapter Triggering Instructions.
2016-04-07	1.0.7	Updated comments for tag 119 and tag 381. Added OpenCloseIndicator field for New Order Single message.
2016-05-17	1.0.8	Addition of new value 5 (replaced) for tag OrdStatus/39 in Order Cancel Reject – Cancel Replace message.
2016-05-23	1.0.9	Changing the description of on close orders.
2016-05-23	1.0.10	Changing tag from 6917(NotionalAmt) to 20015(OptionPremiumAmt) in Execution Report and Trade Capture Report messages.
2016-06-22	1.0.11	Removing the value 3 (Mandatory close) from all tag 20199 (OpenCloseIndicator) occurrences.
		Added chapter on “Multileg Orders”
		Added Security Definition Request and response messages
		Changed "Quantity Conditions" MaxFloor (111) notes
		Added value 2 (Specified Trading Session) to TriggerType/1100 in “Triggering Instructions” section
		Added tag 1113/TriggerTradingSessionID to the “Triggering Instructions” section
		Added Trigger Instructions table in the “Trigger Instructions” section

Date	Revision	Change Description
		Added new section “Triggering off Session Changes”
		Added new section “Open Close Indicator”
		Added new section “On Close Orders”
		Changing tag 452/PartyRole so that “14 = Giveup Clearing Firm” and “83 = Clearing Account” will be supported
		Adding new valid values for tag 528/OrderCapacity
		Modified the tag 386/NoTradingSessions comment
		Added valid values to the comment of tag 336/ TradingSessionID in “New Order Single –inbound to Marketplace (in)” section
		Added new tag 20199/OpenCloseIndicator for NewOrderSingle, OrderCancelReplace and ExecutionReport messages
		Added new tag 119/SettlCurrAmt for ExecutionReport messages
		Added new tag 625/TradingSessionSubID for Acknoledgement ExecutionReports
		Added the comment on identification of the “Execution Report – Unsolicited Cancel (out)”
		Added new tag 21100/LastUnderlyingPx to TradeCaptureReport messages
		Added new section “Trade Capture Report – Deal on Hold Accepted (out) (Currently only supported for drop copy sessions)”
		Removed comment that the “Allocation” messages are not supported
		Removed comment that the “Give-Ups” are not supported
		Modified comments about the tags used to specify the take up account for Trade Capture Report in “Automatic Give-Ups” section
		Added new tag 381/GrossTradeAmt for TradeCaptureReport messages
		Removed “Appendix B, Trade types”
2016-06-23	1.0.12	Updated the Revision History table to have all the document changes added after Phase 1
2016-08-04	1.0.13	Adding PartyRole values 7 and 36 for messages ExecutionReport (out), MassQuoteAcknowledgement (out) and TradeCaptureReport (out)
2016-08-04	1.0.14	Adding chapter 6.5.6 Paused Orders. Adding note to OrderCapacity (528) tag. Modifying chapter 6.6.2 Order States
2016-08-16	1.0.15	Modified the description of supported values for

Date	Revision	Change Description
		tag OpenCloseIndicator(20199) in sections "Open Close Indicator" and "Order Cancel Replace Request (in)". Also removed one allowed value from tag OpenCloseIndicator(20199) in section "New Order Single – inbound to Marketplace (in)"
2016-08-16	1.0.16	Added new comment to "Business Message Reject" section regarding the message count pacing limit.
2016-10-13	1.0.17	Removed tag 1390 from TradeCaptureReport messages.
2016-10-19	1.0.18	Corrected translation of trigger instructions for Best Offer.
2016-10-24	1.0.19	Added information on custom fields and added enumerations. Added tag 20013 to Security Definition message.
2016-11-09	1.0.20	Removed tag 625 from Execution Reports
2016-11-16	1.0.21	Copyright and disclaimer change
2016-12-02	1.0.22	Removed tags 1938, 1939, 1940 from Security Definition Request (c).
2016-12-21	1.0.23	Update fields for precious metal registration.
2017-01-23	1.0.24	Added "Appendix C, Trade types" section.
2017-01-24	1.0.25	Added 20034 (ComboMatchID) tag to "Execution Report – Combination Order Fill (out)" section. Added description for tag 20015 (OptionPremiumAmt). Corrected 20199 description reference. Added 20200 (LegTrdMatchID) tag to "Execution Report – Combination Order Fill (out)" section. Updated the "5.3 Available Party Roles" section.
2017-01-25	1.0.26	Added new values to "Appendix C, Trade types".
2017-01-27	1.0.27	Made 20034 tag mandatory in "7.5.8 Execution Report – Combination Order Fill (out)" section Moved 20200 tag to correct position in "7.5.8 Execution Report – Combination Order Fill (out)" section. Added 20034 tag to "10.4.1 Trade Capture Report – auto-matched trade (out)" section
2017-03-30	1.0.28	Adding additional fields for Security Definition and Security Definition Update messages
2017-09-11	1.0.29	Added new values to "Appendix C, Trade types".
2018-02-27	1.0.30	Added valid value of TradingSessionID to the Execution Reports.
2018-05-07	1.0.31	Updated field description (21101 UnderlyingColPx)
2018-06-16	1.0.32	Added following Trdypes to the "Appendix C, Trade types" table;

		821,831,832,833,841,842,843,844
December 6, 2018	1.0.33	Added following Trdypes to the “Appendix C, Trade types” table; 685, 686 ,687
December 18, 2018	1.0.34	Added “Appendix D, Secondary trade types” section.
February 12, 2019	1.0.35	• Added new comment to section 7.3.5 section regarding number of Execution Report – Combination Order Fill (out) due to number of trade legs. • Section 4.4 is updated. • Section 3.14.4 is updated.
March 26, 2019	1.0.36	• User Request and User Response messages are updated with new functionality. • Adding new valid values for tag 528/OrderCapacity
April 18, 2019	1.0.37	• Added new values to tag 1409 for Logout message(MsgType=5) • Removed value 8 from tag 1409 for Logon message(MsgType=A)
30 April, 2019	1.0.38	• Added new values to tag 1409 for Logout message(MsgType=5)
06 May 2019	1.0.39	• Valid values of tag 926(User Response(out) message) are changed in Section 6.4.2. • Tag 924 (User Response(out) message) is added in Section 6.4.2.
20 May 2019	1.0.40	• Added information about derivatives market in Sections 8.5 and 8.11.4
22 October 2019	1.0.41	• Added new values (27, 28) to tag 21106 for Trade Capture Report (MsgType = AE)