BTS2 FIX SPECIFICATION

ORDER MANAGEMENT

Version	1.16
Date	11 September 2023
File	BTS2 FIX Specification - Order Management v1-16.docm

Document History

VERSION	DATE	SUMMARY OF CHANGES
1.00	2013-01-25	First release of BTS2 FIX Specification: Order Management
1.01	2013-03-01	• Section 3.3.1: Removed a few attributes from the list of order attributes allowed to change. • Table 11: Added new rows for clarity purpose. • Table 12: Added new rows for clarity purpose. • Table 12: Component block <TriggeringInstruction> removed. • Table 12: ExpireDate removed. • Table 12: ExpireTime removed. • Table 17: Added comments • Table 17: MarketID removed. • Section 4: Added • Add SecuirtyGroup (1151) to B.1 Instrument Component Block. • Add Tag 342, 345 and Instrument component block to Trading Session Status(h) .Add comments for Trading Session Status (h). • Add "G – Trade Correct" in ExecType (150) values. • Update Tag 1(Account) is required
1.02	2013-03-11	• Section 3.9 Table 11: Removed PartyRole 4 • Section 3.9 Table 11: Added value 'M' (Market Maker) for OrderCapacity (528) • Section 3.10 Table 12: Removed PartyRole 4 • Section 3.12 Table 14: Removed PartyRole 4 • Section 3.12 Table 14: Amended comment on Account (1) • Section 3.12 Table 14: Removed OrderCapacity (528)
1.03	2013-03-22	• Section 3.9 Table 11: Removed Yield(236), ExpireDate(432), ExpireTime(126), DisplayQty(1138) • Section 3.11 Table 13 : Removed Component block <Parties>, Component block <Instrument>, OrderQty • Section 3.12 Table 14: Removed Yield(236), ExpireDate(432), ExpireTime(126), DisplayQty(1138) • Section 3.14 Table 16: Removed Component block <Parties>, Component block <Instrument> • Section 3.15 Table 17: Removed Component block <Parties> • Section 3.16 Table 18: Removed Component block <Parties> • Section 3.17 Table 19: Added SecondaryOrderID(198); Added comment to SecondaryCIOrdID(526) • Section 4.3 Table 21: Added QuoteType(537) • Section 4.3 Table 21: Removed QuoteSetValidUntilTime (367), ValidUntilTime(62)

VERSION	DATE	SUMMARY OF CHANGES
1.04	2013-04-15	• Change the maximum length of Text (58) and PartyID (448) when PartyRole = ClientID to 24 characters • Change OrderRestrictions(529) to required and the maximum length of OrderRestrictions (529) is 5 characters • Change Text (58) to not required in Allocation Instruction Ack (P)
1.05	2013-04-19	• Added section 2.2 – FIX Session establishment scenarios • Update valid value for OrderRestrictions (529)
1.06	2013-05-20	• Add Section 1.5 FIX Compression • Updated description for tag 871 – InstrAttribType • Removed PendingNew from OrdStatus (39) and corrected comments for PendingCancel (6) in ExecType (150) in Appendix C. • Change password maximum length to 12 characters. • Updated Figure 9, 13, 14, 16, 17, 18 • Add 'CBH' to Tag 336 TradingSessionID. • Removed MassQuote can be amened by OrderCancelReplaceRequest. • Section 3.10 Table 12: Correct MsgType=s • Section 6.4 Table 26: Correct OrderRestrictions(529) = In/Out • Section 6.4 Table 26: Update component block<Parties>, PartyRole(11) is mandatory • Tag TrdType (828) is changed to not required in Trade Capture Report. • Correct item 7 in 5.2.3 - Workflow for One-Party Report for Pass-through to Counterparty.
1.07	2013-06-18	• In Section 5.4 - Trade Capture Report (AE), two sided trade capture report requires counterparty ExecutingTrader (12) only. • Added Parties component block to Allocation Instruction (J). • Add withdraw on logoff quote types to MassQuote and withdraw on log off value in ExecInst (18). • Add limit of 12 QuoteEntryGrps in a QuoteSetGrp in MassQuote (i). • Changed Tag 871 =111 to "Delivery Basis".
1.08	2013-09-03	• In Table 20 – Execution Report Returned Tags Based On Scenario, deleted tag 103 OrdRejReason in row Order Cancel/Replace Reject. • Deleted incorrect statement in section 2.1.4 - Changing FIX Session Passwords. • Tag Price (44) represents clean price for yield based products in Table 11 – New Order Single and Table 14 – Order Cancel/Replace Request. • Tag GrossTradeAmt (381) includes accrued Interest for convertible bonds and fixed income products. • Added Tag AccruedInterestAmt (159) to Excecution Report (8). • Removed Tag StrikeCurrency (947) from Instrument component. • Removed Tag YieldRedemptionDate (696) from YieldData component.

VERSION	DATE	SUMMARY OF CHANGES
1.09	2014-01-27	• Add in techical guide
1.10	2014-06-18	• Updated : Support Fill or Kill(59=4) • Updated : Support Good till date(59=6) • Added : ExpireDate(432) • Added : ExecRestatementReason(378)
1.11	2016-12-05	• Appendix B.Table 31: SecurityGroup (1151) added 'LEAP'.
1.12	2017-10-06	• Updated : Support Good till cancelled(59=1)
1.13	2018-02-05	• Added : Support IDSS(54=I)
1.14	2018-08-23	• Added : Support At the Opening(59=2) • Added : Support At the Close(59=7) • Added : Support One Cancel Other(OCO) order option : Add tag ClOrdLinkID(583) • Added : Support DisplayQty(1138) : Iceberg • Added : Support Session(59=S)
1.15	2019-10-10	• Added : Support Permitted Short Sell(PSS) for MassQuote and New Order Single
1.16	2023-09-11	• Updated : MassActionScope 7 and 9 are not supported when MassActionType=3 for Order Mass Action Request

Contents

Figures

Tables

1 FIX Specification for BTS2

This document provides the BTS2 FIX message specification supporting version 5.0 (SP1) of the FIX protocol specification.

1.1 FIX V5.0 (SP1) Supported Messages

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

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

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

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

1.2 Document Structure

This document is divided into the following chapters:

Chapter 2 – Session and Infrastructure messages.

Chapter 3 – Order Management

Chapter 4 – Quote Management

Chapter 5 – Account Modification

Chapter 6 – Trade Capture Reporting

Chapter 7 – Trade Cancellation

Chapter 8 – Market & Reference Data

Appendix A describes the FIX Standard Header and Standard Trailer.

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

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

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

1.3 References

The following references are documents related to this specification.

Table 1 – Reference Documents

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

1.4 FIX Messages Supported

The following table lists the FIX messages that are supported by the FIX Gateway. The messages and securities that can be accessed by FIX connections as configured by the Exchange.

Table 2 – FIX Messages Supported

MESSAGE NAME	MESSAGE TYPE	MESSAGE DIRECTION	MESSAGE FUNCTION
Logon	A	Inbound Outbound	Identifies and authenticates a user/member establishing a connection to the gateway.
Logout	5	Inbound Outbound	Used to terminate a FIX session.
Reject	3	Inbound Outbound	Response message providing notification regarding messages that cannot be processed by the gateway or FIX Client.
Resend Request	2	Inbound Outbound	Initiates a re-transmission of messages from the gateway.
Sequence Reset (Gap Fill)	4	Inbound Outbound	Message has two modes: Sequence Reset - Gap Fill and Sequence Reset-Reset.
Test Request	1	Inbound Outbound	Verifies sequence numbers or communications line status.
Heartbeat	O	Inbound Outbound	Monitors gateway status during periods of inactivity.
Business Message Reject	j	Outbound	Rejects any application message that cannot be processed by the Gateway and cannot be rejected via another message.
New Order - Single	D	Inbound	Used by institutions wishing to electronically submit securities orders for execution.
New Order Cross	s	Inbound	Used to submit a cross order into a market

MESSAGE NAME	MESSAGE TYPE	MESSAGE DIRECTION	MESSAGE FUNCTION
Order Cancel Request	F	Inbound	Request to cancel all of the remaining quantity of an existing order.
Order Cancel / Replace Request	G	Inbound	Request message to change the details of an existing order.
Order Cancel Reject	9	Outbound	Reject message for an Order Cancel / Replace Request or Order Cancel Request that cannot be honoured.
Order Status Request	H	Inbound	Request for querying the details of an order.
Order Mass Action Request	CA	Inbound	Used to request the cancellation or status of a group of orders that match the criteria specified in the request.
Order Mass Action Report	BZ	Outbound	Acknowlegdement to an Order Mass Action Request.
Execution Report	8	Outbound	Responds with the action BTS2 has taken in response to a new or existing order including acknowledges Order Cancel and Cancel / Replace Requests, order history requests and report fills to orders. Fills against orders are reported via the Execution Report message, as are trade cancels.
Mass Quote	i	Inbound	Used by market makers to submit market maker order/quotes for multiple securities.
Mass Quote Acknowledgment	b	Outbound	Mass Quote Acknowledgement is used as the application level response to a Mass Quote message.
Allocation Instruction	J	Inbound	Used to execute client amendment, which is an order-level modification on CDS account of an order and all its related trades.
Allocation Instruction Ack	P	Outbound	Used to acknowledge to an Allocation Instruction.
Trade Capture Report Request	AD	Inbound	Request all firm trades and to subscribe or unsubscribe for trade capture reports.
Trade Capture Report	AE	Inbound Outbound	Responds to a Trade Capture Report Request and used to report matched trades, DBT (direct business transations)
Trade Capture Report Request Ack	AQ	Outbound	Used to indicate if no trades matched the selection criteria specified in the Trade Capture Report Request or the Trade Capture Request was invalid.

MESSAGE NAME	MESSAGE TYPE	MESSAGE DIRECTION	MESSAGE FUNCTION
Trade Capture Report Ack	AR	Outbound	The Trade Capture Report Ack message can be: • Used to acknowledge trade capture reports received from a counterparty • Used to reject a trade capture report received from a counterparty
Trading Session Status Request	g	Inbound	Request information on the status of a market.
Trading Session Status	h	Outbound	Responds with the current status of a market.
Security Status Request	e	Inbound	Requests the status of a security. One or more Security Status message are returned as a result of a Security Status Request.
Security Status	f	Outbound	Responds with the current state of a security that is currently listed. May be filtered by board.

1.5 FIX Compression

X-stream FIX offers optional compression on the FIX session. The compression algorithm used is the well-known and widely used ZLIB compression.

ZLIB implementations are available in a wide range of free, open source and commercial tools (zlib.net).

Programming languages such as Java have built-in ZLIB implementations.

If configured, all messages outbound from the marketplace are compressed. Inbound messages should NOT be compressed.

The following Java code sample illustrates the decompression process:

```
Inflater inflater = new Inflater(); //Inflater for decompress
// compressed input stream
InputStream inputStream = clientSocket.getInputStream();
byte buf[] = new byte[1024]; //buffer for reading compressed bytes
byte decompressed[] = new byte[1024]; //buffer for decompressed bytes
while (clientSocket.isConnected()) {
    while (!inflater.finished()) {
           //output byte array stream for assemble decompressed bytes
           ByteArrayOutputStream baos = new ByteArrayOutputStream();
           if (inflater.needsInput()) {
                  len = inputStream.read(buf);
                  if (len == -1)
                         break; //handle EOF
                  inflater.setInput(buf, 0, len);
                  }
           int decompressedLen = inflater.inflate(decompressed);
           baos.write(decompressed, 0, decompressedLen);
    }
    //do normal processing
    process(baos. toByteArray());

    int remaining = inflater.getRemaining();
    //reset inflater for next compression unit
    inflater.reset();
    if (remaining > 0) {
           //put the remaining in inflater for next compression unit
           inflater.setInput(buf, len - remaining, remaining);
    }
}
```

2 Session and Infrastructure Messages

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

The FIX Session Level messages are:

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

The FIX Infrastructure messages are:

- Business Message Reject

2.1 FIX Session Establishment

2.1.1 Logon and Authentication

A FIX session must be established with BTS2 before the exchange of business messages is allowed. The session is established using the Logon message and part of session establishment processing includes the authentication of the initiator. This requires that a valid SenderCompID (49) which defines the party initiating the session, and a password, is provided in the Logon message which can be used for security authentication purposes. A FIX session will not be established if authentication processing fails.

2.1.2 Logon Failures and Account Locking

All logon failures return a Logout message with an appropriate reason code and may include additional text which provides additional information regarding the failure. If the session initiator fails to authenticate with the BTS2 system within a defined number of attempts[default is 3 times], the account will be locked and all subsequent logon attempts will be rejected. If logons are disabled by the marketplace, a failure to logon will not cause the account to be locked but only rejected. Any other causes for authentication failure will cause the account to be locked after a defined number of failed attempts. To unlock the account requires marketplace operations to reset the account and assign a new password.

2.1.3 FIX Session SenderCompID, Username and Passwords

The SenderCompID (49), username (553) and session password are always required for authentication with BTS2 and must be included in the Logon message. Both SenderCompID and Username (553) can have a maximum length of 30 characters. Session password (when in plain text) can have a maximum length of 12 characters.

2.1.4 Changing FIX Session Passwords

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

- Passwords used for X-stream session authentication must be changed on a periodic basis as they have a limited lifetime. Passwords can be changed programmatically using the Logon message only at session establishment and only while they are valid. If the password has expired or cannot be changed programmatically due to limitations of the Participant's FIX implementation then they must be changed manually by Business or Technical Operations.
- To change the FIX session password at logon time both the current password and the new password must be included in the FIX Logon message. In addition, the SenderCompID (49), UserName (553) and the current password must be valid otherwise authentication will fail.
- The current password is sent using the Password (554) field in the Logon message. It should be noted that the password will transit external and internal X-stream networks in plain text if encryption is not utilized (refer to Section 2.1.5 regarding encryption).
- Providing the SenderCompID, UserName (553) and current password are valid, the new password is checked against the password policy for compliance. If the new password complies, it is updated in the X-stream database and becomes the password to be used for the next session logon. If the new password does not comply with the password policy then an error status and message is returned in the Logon confirm message.

2.1.5 Encryption

FIX for BTS2 does not currently support either password or message encryption over FIX sessions. If encryption is required then hardware based encryption must be used.

2.1.6 FIX Session Logon Confirmation and Logout

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

2.1.6.1 Logon Confirmation (Session Authenticated)

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

2.1.6.2 Logout (Authentication Failure)

Failure to establish a session with BTS2 for any reason will return a Logout message. The Text(58) field in the Logout message may contain additional useful information regarding the reason for the Logout message being returned.

2.1.7 FIX connections and broker codes

The FIX connections format provided below are ONLY APPLICAPLE to BTS2 Production platform. These format are not applicable in the BTS2 Certification(UAT) platform.

	Order Types	Broker Code to Use	Example
1	*Orders submitted to • Normal Market • Odd-Lot Market • Buy-In Market	Every order must be submitted with a Broker Code, where the Branch Code begins with '9'. This broker code is issued along with the FIXTRADER connection.	• **012901** • **034902** • **055903** • **068904**
2	Market maker orders submitted to • Normal Market	Every order must be submitted with a Broker Code, where the Branch Code begins with '1'. This broker code is issued along with the FIXTRADER connection.	• **012101** • **034101** • **055102** • **068102**
3	Orders submitted to • DBT Market	Every order must be submitted with a Broker Code, where the Branch Code begins with '2'. This broker code is issued along with the FIXNEGDEAL connection. Note: When negotiating DBT with counterparty, ensure Broker Codes are provided in the right format by both parties of the trade.	• **012201** • **034202** • **055201** • **068202**

	FIX Connection Type	Description	Associated Broker Code Format
1	**FIXTRADER**	1. Orders accepted for the following board: • Normal • Odd-Lot • Buy-In 2. Also accepted are Market Maker orders in the Normal board.	1. Format = XXX9XX • 6-digit with the 4th digit being '9' • eg. 068901, 033902, 076905. 2. Format = XXX1XX • 6-digit with the 4th digit being '1'. • eg. 068101, 033102, 076101.
2	**FIXNEGDEAL**	1. Orders accepted for the following board: • Direct Business Transactions, Off-Market	1. Format = XXX2XX • 6-digit with the 4th digit being '2'. • eg. 068201, 033202, 076201.

2.2 FIX Session establishment scenarios

There are four types of FIX session establishment scenarios:

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

2.2.1 Start of the day

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

2.2.2 Reconnect

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

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

FIX message subscriptions submitted before disconnection will still generate messages during disconnection, same as execution reports. These messages can be retrieved using the Resend Request message.

For Trade Capture Report only latest snapshots will be provided. Example, if the Trade Capture Report is confirm[matched] and cancel [TradeCancel] by the counterparty during disconnection then only one Trade Capture Report[TradeCancel] is provided after reconnected.

2.2.3 Fail over

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

2.2.4 Disaster recovery

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

2.2.5 Fail over scenario and strategy

	Failover Type	Description
1	Main Site Component Failover	Primary FIX Gateway fail to Secondary FIX Gateway on Bursa Malaysia Main Site
2	DR Site Component Failover	Primary FIX Gateway fail to Secondary FIX Gateway on Bursa Malaysia DR Site
3	Main to DR Site Failover	Primary/Secondary FIX Gateway in Bursa Main Site, fail to Primary FIX Gateway on Bursa Malaysia DR Site

Scenario 1 & 2 : Component Failover

		Impact
1	Target IP Address & Port (BTS2 FIX Gateways)	• The IP address of the secondary BTS2 FIX Gateway differs from the primary gateway. However, port configurations remain identical for both gateways. In summary, target IP address must be changed to point to the secondary gateway but not the port numbers.
2	FIX Username, Password, SenderCompID & TargetCompID	• FIX IDs will remain the same. This is because both the primary and secondary gateways share the same repository.
3	Trades & Orders	• Trades and outstanding orders remain valid and accessible via secondary gateway.
4	Market Data subscription	• All subscriptions for incremental updates will be dropped. Upon connecting to the Secondary BTS2 FIX gateway, new subscriptions are required in order to receive snapshot/incremental market data.
5	Trade Capture Report subscription	• Trade Capture Report subscriptions will be dropped. Upon connecting to the Secondary BTS2 FIX gateway, new subscriptions are required in order to receive Trade Capture Reports.
6	Incoming & Outgoing Sequence Numbers	• Applicable to FIXTRADER, FIXNEGDEAL, FIXMDFULL, DROPCOPY or any FIX connection. • Connect to secondary gateway with the NEXT SEQUENCE NUMBERs. If incoming sequence number is higher than expected upon connecting to the secondary server, send "RESEND REQUEST" to request for the missing messages.

Component failover is a one-way failover. Once the primary FIX gateway fails to the secondary gateway, the secondary gateway will stay operational till the end of the day. The primary FIX gateway will be back in service the following day.

Scenario 1 & 2 : Failover Strategy

	FIX Connection Type	Approach
1	FIXTRADER	Follow this sequence: 1. Connect to secondary BTS2 FIX Gateway with next sequence numbers 2. Send any FIX subscription if it was sent earlier to the primary BTS2 FIX Gateway.
2	FIXNEGDEAL	Follow this sequence: 1. Connect to secondary BTS2 FIX Gateway with next sequence numbers 2. Send the Trade Capture Report Request.
3	DROPCOPY	Follow this sequence: 1. Connect to secondary BTS2 FIX Gateway with next sequence numbers 2. Send any FIX subscription if it was sent earlier to the primary BTS2 FIX Gateway.
4	FIXMDFULL or any other FIX market data connection	Follow this sequence: 1. Connect to secondary BTS2 FIX Gateway with next sequence numbers 2. Send any FIX subscription if it was sent earlier to the primary BTS2 FIX Gateway.

Scenario 3: Main to DR Site Failover

		Impact
1	Target IP Address & Port (BTS2 FIX Gateways)	• The IP address of the DR-Site BTS2 FIX gateway differs from the Main-Site. However, port configurations remain identical. In summary, target IP address must be changed to point to the DR-Site BTS2 FIX Gateway but not the port numbers.
2	FIX Username, Password, SenderCompID & TargetCompID	• FIX IDs will remain the same. This is because the Main-Site configurations are replicated to the DR-Site.
3	Trades & Orders	• Trades and outstanding orders remain valid and accessible via DR-Site BTS2 FIX gateway.
4	Market Data subscription	• All subscriptions for incremental updates will be dropped. Upon connecting to the DR-Site BTS2 FIX gateway, new subscriptions are required in order to receive snapshot/incremental market data.
5	Trade Capture Report subscription	• Trade Capture Report subscriptions will be dropped. Upon connecting to the DR-Site BTS2 FIX gateway, new subscriptions are required in order to receive Trade Capture Reports.
6	Incoming & Outgoing Sequence Numbers	• Applicable to FIXTRADER, FIXNEGDEAL, FIXMDFULL, DROPCOPY or any FIX connection. • Connect to the DR-Site BTS2 FIX gateway with the BOTH SEQUENCE NUMBERs = 1. DO NOT SEND LOGON with ResetSeqNumFlag. If incoming sequence number is higher than expected upon connecting to the secondary server, send "RESEND REQUEST" to request for the missing messages.

Scenario 3: Failover Strategy

	FIX Connection Type	Approach
1	FIXTRADER	Follow this sequence: 1. Connect to DR-Site BTS2 FIX Gateway with both incoming and outgoing sequence numbers equal to "1". DO NOT SEND LOGON with ResetSeqNumFlag. 2. The incoming sequence number would probably be higher than 1 because all the Execution Reports received in Main-Site BTS2 FIX Gateway are replicated to the DR-Site. Upon connection, and if the incoming sequence number is 2001, as an example; send a RESEND REQUEST for missing messages from 1 to 2000, and all the Executions Reports will be replayed. 3. Wait for the completion of receiving all the missing messages. 4. Send any FIX subscription if it was sent earlier to the Main-Site BTS2 FIX Gateway.
2	FIXNEGDEAL	Follow this sequence: 1. Connect to DR-Site BTS2 FIX Gateway with both incoming and outgoing sequence numbers equal to "1". DO NOT SEND LOGON with ResetSeqNumFlag. 2. Send the Trade Capture Report Request. Upon receiving this request, the DR-Site BTS2 FIX gateway will replay all the statuses of Trade Capture Reports.
3	DROPCOPY	Follow this sequence: 1. Connect to DR-Site BTS2 FIX Gateway with both incoming and outgoing sequence numbers equal to "1". DO NOT SEND LOGON with ResetSeqNumFlag. 2. Upon connection, and if the incoming sequence number is 2001, as an example; send a RESEND REQUEST for missing messages from 1 to 2000, and all the Executions Reports and all the Trade Capture Reports received by FIXTRADER and FIXNEGDEAL connected to the BTS2 DR-Site FIX gateway will be replayed.
4	FIXMDFULL or any other FIX market data connection	Follow this sequence: 1. Connect to DR-Site BTS2 FIX Gateway with both incoming and outgoing sequence numbers equal to "1". 2. Send any FIX subscription if it was sent earlier to the Main-Site BTS2 FIX Gateway.

2.3 Logon (A)

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

Table 3 – Logon

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
Standard Header		Y	MsgType = A	
98	EncryptMethod	Y	(Always unencrypted)	Int
108	HeartBtInt	Y	Note same value used by both sides	Int
141	ResetSeqNumFlag	N	Indicates both sides of a FIX session should reset sequence numbers	Boolean
1137	DefaultApplVerID	Y	Specifies the service pack release being applied by default to the message at the session level. The only valid value is '8' = FIX50SP1.	String
553	Username	Y	Specifies a different username or userID to use for authentication	String
554	Password	Y	Note: minimal security exists without transport-level encryption.	String (12)

TAG	FIELD NAME	REQ'D	COMMENTS	FORMAT
925	NewPassword	N	Specifies a new password when required. Maximum length 12 characters.	String (12)
58	Text	N	Free format text string	String
Standard Trailer		Y		

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

2.4 Logout (5)

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

The logout format is as follows.

Table 4 – Logout

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

2.5 Reject (3)

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

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

The reject format is as follows.

Table 5 – Reject

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

2.6 Resend Request (2)

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

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

The resend request format is as follows.

Table 6 – Resend Request

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

2.7 Sequence Reset (Gap Fill) (4)

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

Gap Fill mode

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

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

Reset mode

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

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

Table 7 – Sequence Reset

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

2.8 Test Request (1)

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

The TestReqID verifies that the opposite application is generating the heartbeat as the result of Test Request and not a normal timeout. The opposite application includes the TestReqID in the resulting Heartbeat. Any string can be used as the TestReqID (one suggestion is to use a timestamp string). The test request format is as follows.

Table 8 – Test Request

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

2.10 Heartbeat (0)

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

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

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

The heartbeat format is as follows.

Table 9 – Heartbeat

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

2.11 Business Message Reject (j)

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

Table 10 – Business Message Reject

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = j (lowercase)	
45	RefSeqNum	N	MsgSeqNum of rejected message	SeqNum
372	RefMsgType	Y	The MsgType of the FIX message being referenced.	String
379	BusinessRejectRefID	N	The value of the business-level "ID" field on the message being referenced. Required unless the corresponding ID field (see list above) was not specified.	String
380	BusinessRejectReason	Y	Code to identify reason for a Business Message Reject message. Code to identify reason for a Business Message Reject message.	Int
58	Text	N	Free format text string	String
Standard Trailer		Y		

3 Order Management

The order management category consists of the following messages:

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

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

3.1 Unique ClOrderId (11)

BTS2 **will not check for uniqueness of ClOrdId(11)** on New Order Single, Order Cancel/Replace Request and Order Cancel Request messages. Firms submitting order transactions via FIX interface must ensure unique ClOrdId(11) is entered on these transactions.

When an action (order modification or order cancellation) is requested on a ClOrderId that happens to be duplicated, only the last order identified by ClOrderId is affected.

3.2 Order Identification

A FIX order is identified by either by its current ClOrderId using OrigClOrdID (41), or by BTS2 OrderID (37).

If BTS2 OrderID (37) is used, OrigClOrdID(41) should be set to "NONE". OrderID (37) is unique for every order. Note that OrderID (37) can be renumbered after order amendment.

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

A new ClOrderId must be provided in the Order Cancel/Replace Request message.

3.3.1 Order Attributes allowed to change

Although the FIX protocol allows for virtually all of the Order attributes to be changed, there are limitations as to what the back-end BTS2 system allows. Attributes that are not allowed to change, even if they are mandatory in FIX protocol, are not validated. The following attributes are allowed to change:

- OrderQty (38)
- Price (44)
- OrdType (40)
- TimeInForce (59)
- ExpireDate(432)
- ExecInst (18)
- TriggerPrice (1102)
- OrderRestrictions(529)
- Text(58)
- DisplayQty(1138)

Note: Any change to the price of an order, or increasing quantities will result in the order losing its priority in the market. Trading firms are to ensure that other type of modifications, other than the above, are disallowed/blocked.

3.4 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 unsolicited Execution Reports to the client.
- The system will generate cancel messages (Execution Report –IOC/Fok Order Cancel) for every IOC and FoK order.

3.5 Cross Protocol Order Management

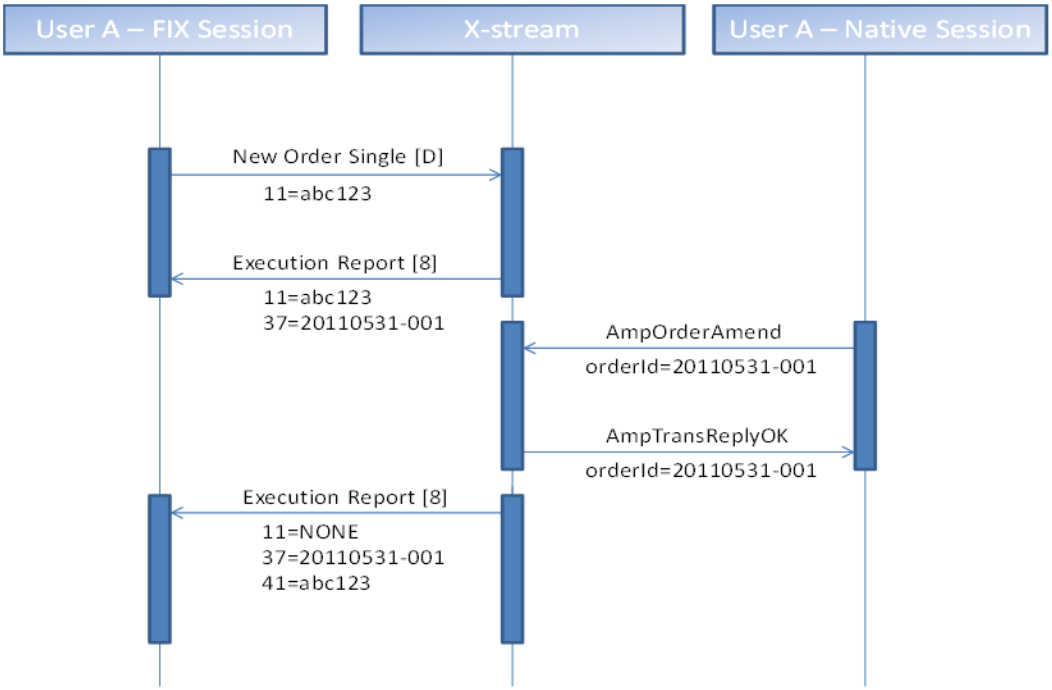
All orders entered via the native protocol will be published as Execution Reports on the FIX session(s) for the same member. Please note that no ClOrdID will be set on those Execution Reports. These orders can be replaced, cancelled or status requested via FIX. System-generated OrderID (37) will be available.

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

3.5.1 Orders Entered via FIX

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

Figure 1 – FIX Order Amended via Native Protocol

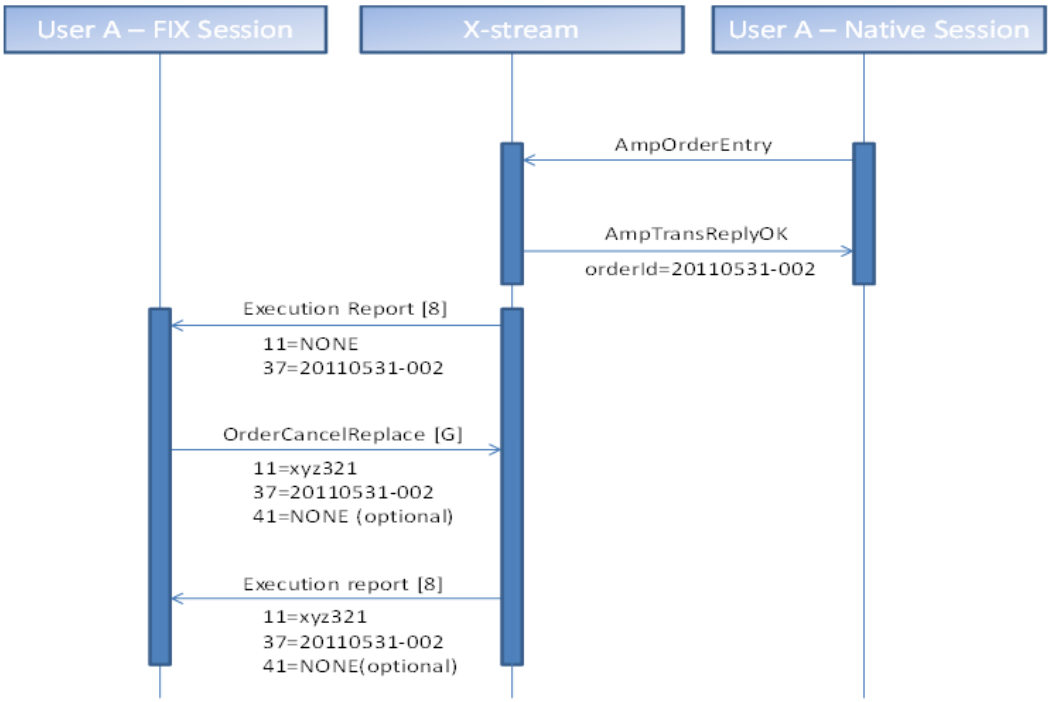


If the order is amended twice by a native client, then 41 will not be present in the ExecutionReport (8)

3.5.2 Orders Entered via Native Protocol

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

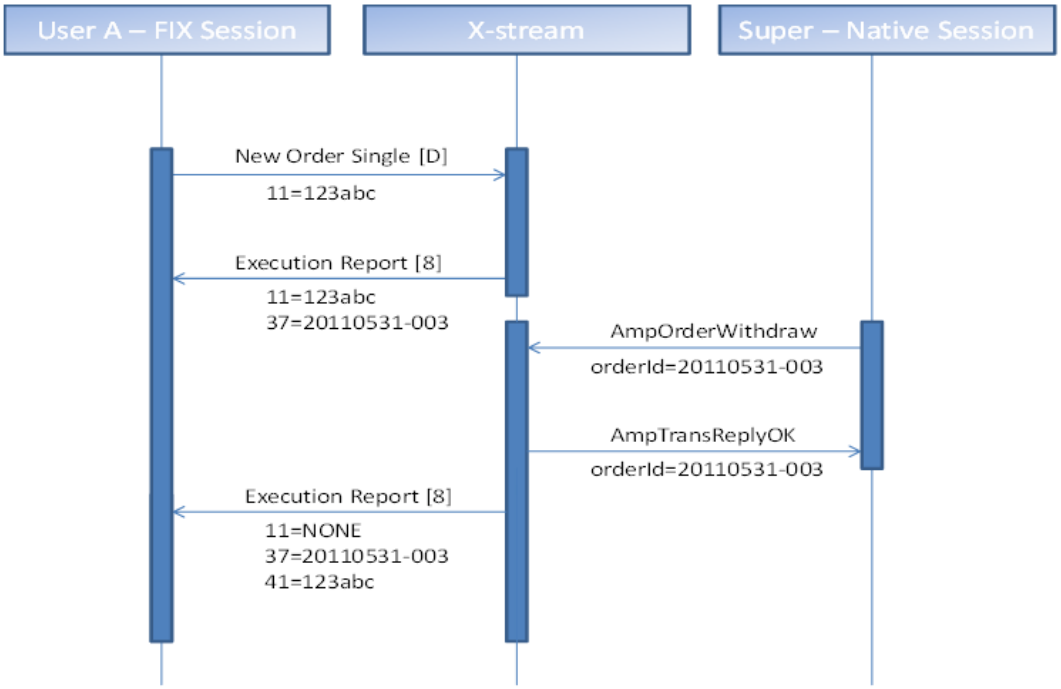
Figure 2 – Order Entered Via Native Protocol and Amended via FIX



3.5.3 Supervisor Cancellation of Orders

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

Figure 3 – FIX Order Cancelled by Supervisor



3.6 On-Behalf Order Management

FIX sessions are mapped directly one-to-one to BTS2 native TCP/IP sessions.

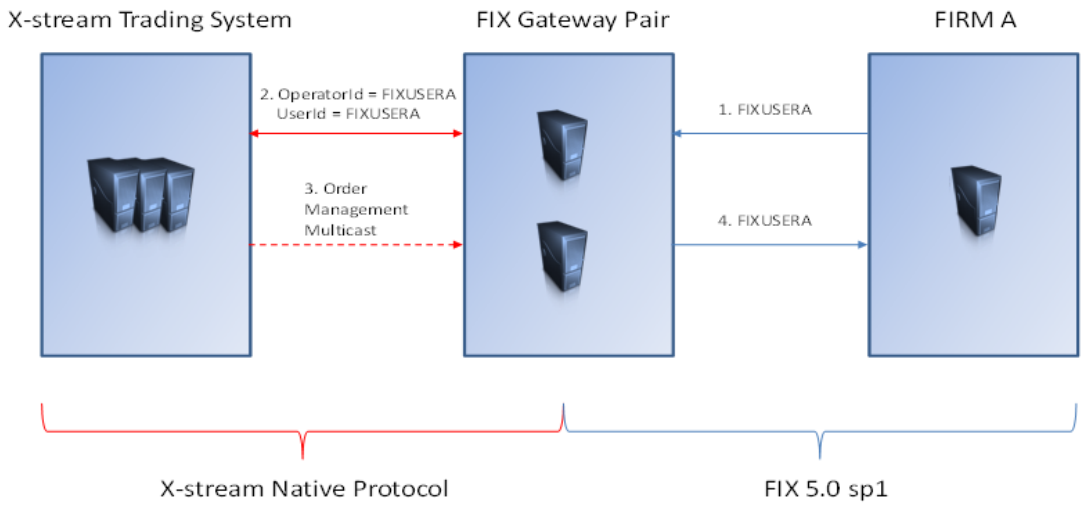
Once a FIX connection is established, the Logon (A) message initiates a native session with the BTS2 trading environment.

The Username (553) and Password (554) are used to authenticate with the BTS2 back-end. This Username (553) will then be used as the operator identifier for this session to BTS2.

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

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

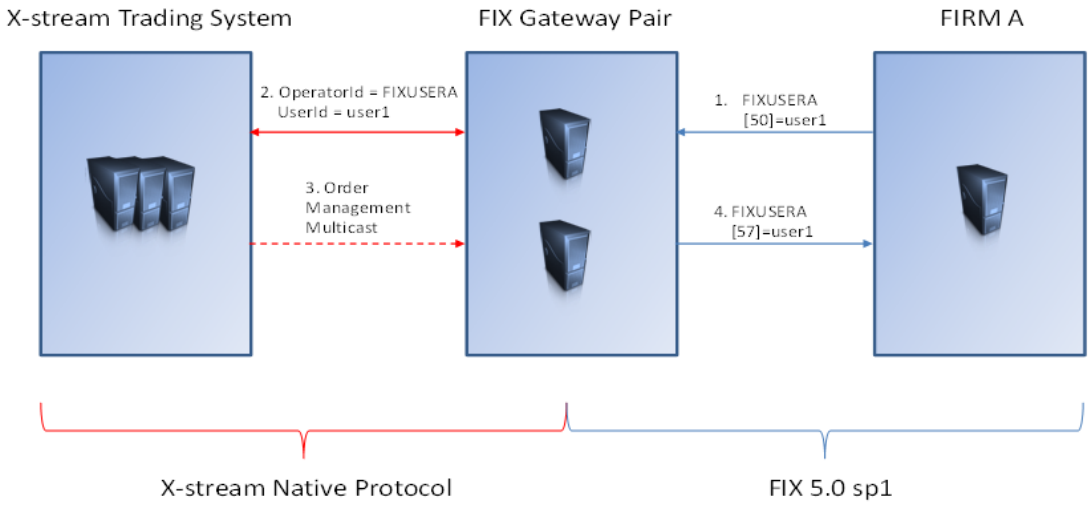
Figure 4 – Message Flow for FIX without SenderSubID (50)



A FIX session will map one-to-one with a native BTS2 (X-stream) session. A FIX user may send order management messages without a SenderSubID (50).

The FIX gateway will send the transaction, to the back-end, with the just the Username (553) value from the originating Logon (A) message. The transaction will then be validated in BTS2 (X-stream) using the permissions associated with this user.

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



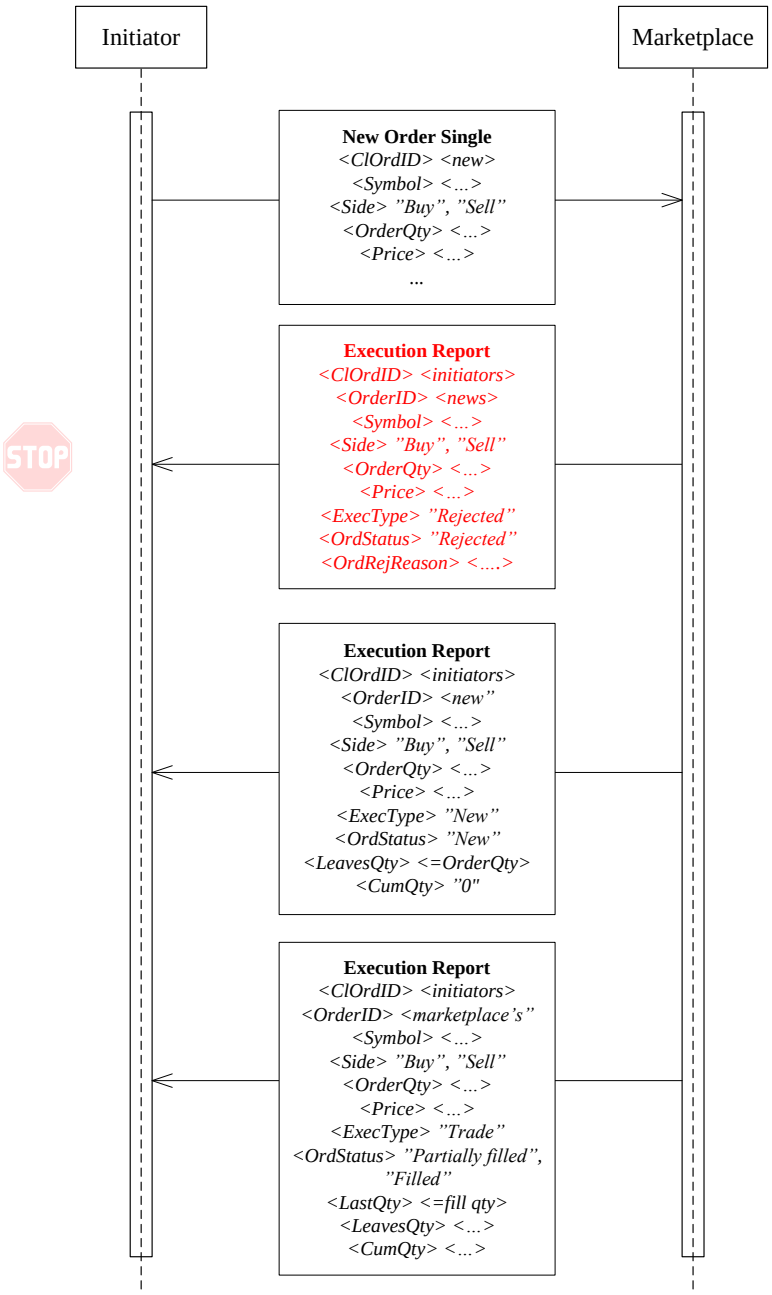
A FIX order message with SenderSubID (50) will send two usernames to the BTS2 (X-stream) backend – OperatorId and UserId. BTS2 (X-stream) first checks that the OperatorId, the owner of the FIX session, has permission to enter messages 'on-behalf' of the UserId from the SenderSubID (50). The transaction is then processed with the permissions of UserId.

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

3.7 Workflows

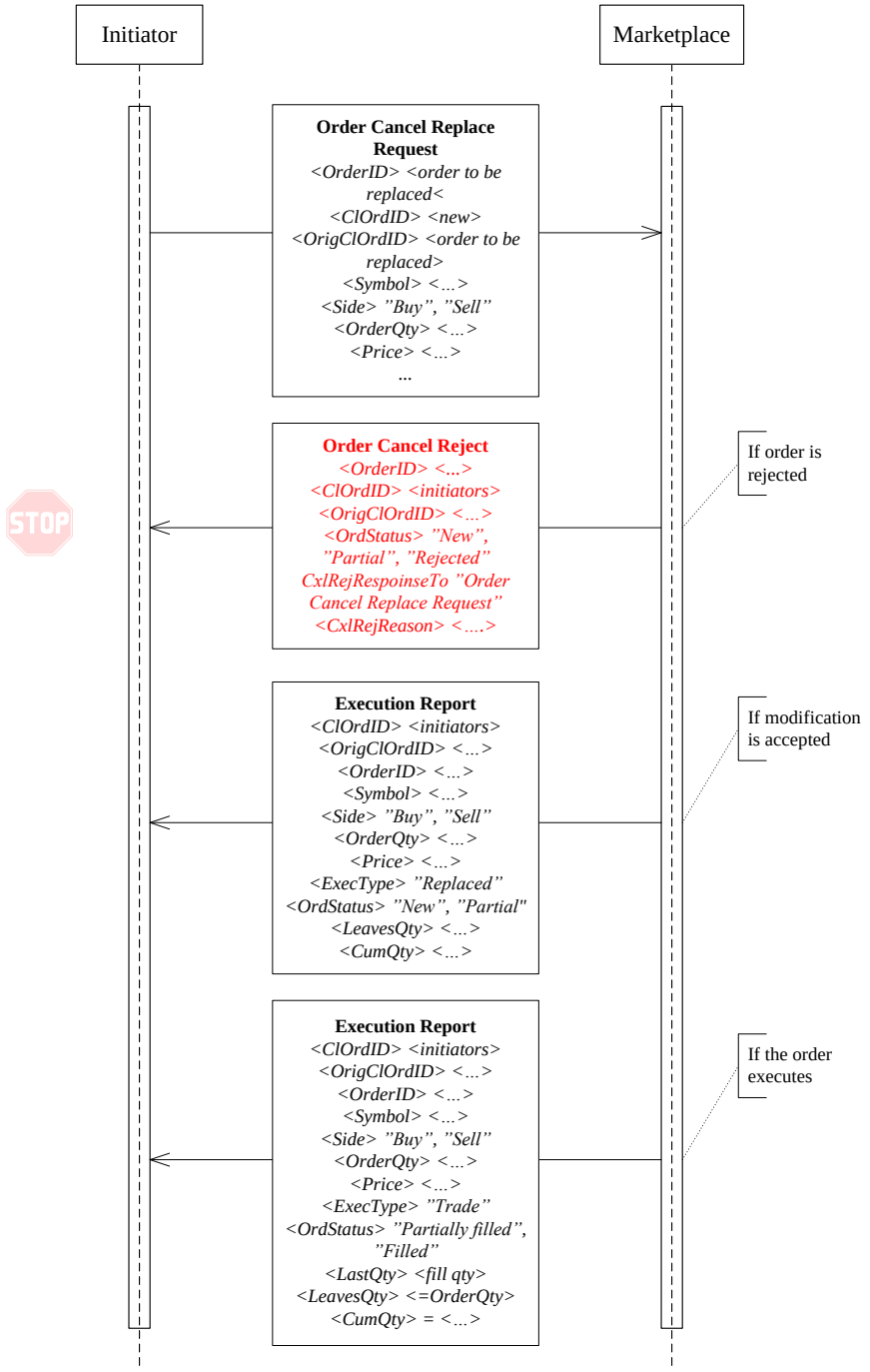
3.7.1 Entering of an New Order

Figure 6 – New Order Entry Workflow



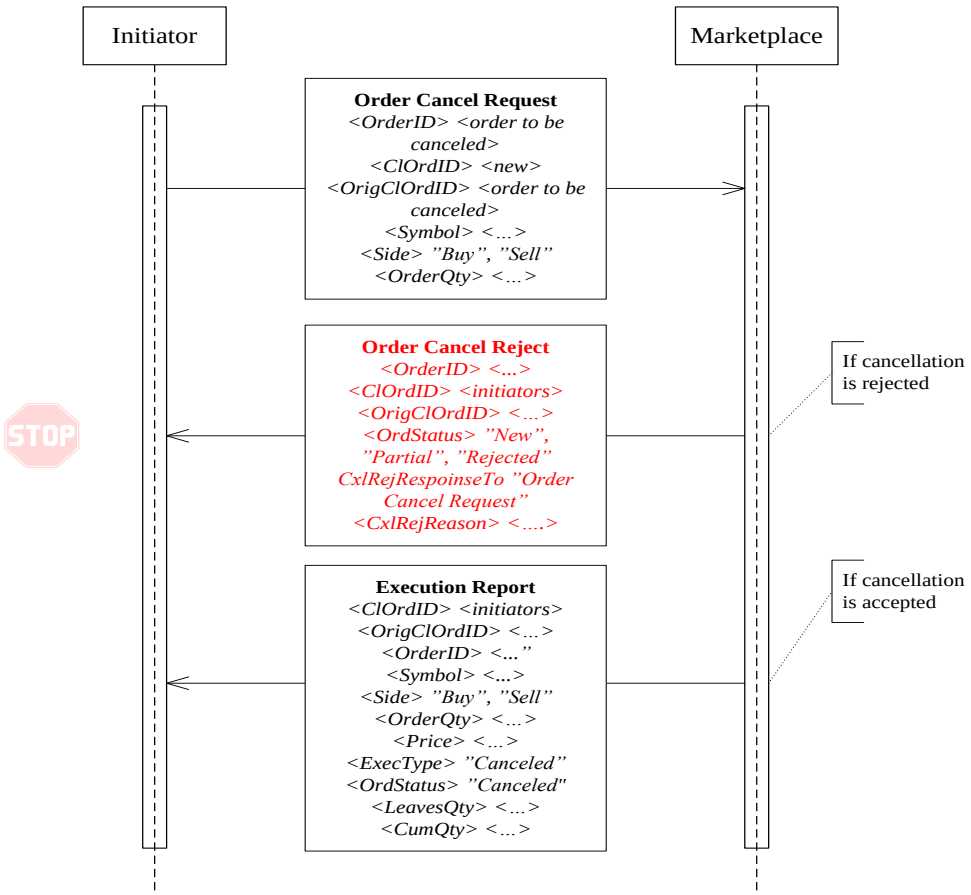
3.7.2 Modification of an Order

Figure 7 – Order Modification Workflow



3.7.3 Order Cancellation

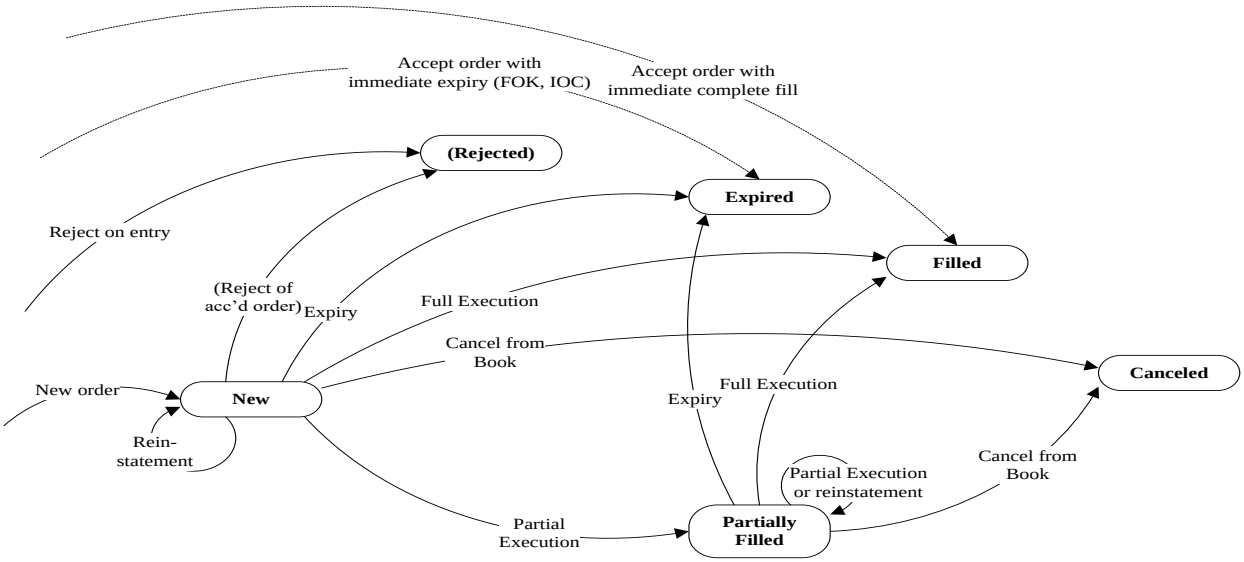
Figure 8 – Order Cancellation Workflow



3.7.4 Order Status

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

Figure 9 – Order Status States



3.8 Crossing Order

3.8.1 Entering of a crossing order

Crossing order is also known as On Market Married Transactions. Entering of a crossing order is accomplished by sending the New Order Cross (s) message.

A unique CrossID (548) must be set in the New Order Cross (s) message. Additionally, each side of the cross order should have a unique ClOrdID (11).

3.8.2 Execution Report of a crossing order

Each successful crossing order will generate two Execution Report (8) messages, one for each side of the cross order. The CrossID (548) in Execution Report (8) will match the CrossID in New Order Cross. The ClOrdID (11) in Execution Report (8) will match the ClOrdID in the same side of New Order Cross.

3.9 New Order Single (D)

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

Table 11 – New Order Single

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = D	
11	ClOrdID	Y	Unique identifier for Order as assigned by firms. See Appendix C.	String (20)
583	ClOrdLinkID	N	Unique identifier to link One Cancel Other(OCO). Valid value: ClOrdID(11) of the previous non-rejected order.	String (20)
Component block <Parties>		Y N	Insert here the set of "Parties" fields. PartyRole 11 (Trader or Dealer ID) PartyRole 3 (Client ID, free to use) See Appendix B.	String (20) String (24)
Start of Component block, expanded in line <Instrument>				
48	SecurityID	Y	Unique marketplace assigned identifier number for an order book. Eg. "1818", "1818WA".	String
22	SecurityIDSource	Y	Identifies class or source of the SecurityID (48) value. Valid value is: 99 – Marketplace assigned identifier	String
762	SecuritySubType	Y	Specifies the board on which SecurityID is listed. This field is equivelant to MarketSegmentID (1300). Valid values are: NM – Normal board OD – Odd-lot board BI – Buy-IN board	String
End of Component block, expanded in line <Instrument>				
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields. See Appendix B.	
1	Account	Y	Specifies Investor Account field. This is the 9-digit CDS account. Left-padded with "0" when required. E.g. "000181818".	String (9)
18	ExecInst	N	Instructions for order handling. Only 'G' option is allowed (All or None). Note, whenever using this value to specify an All or None order, the Minimum Quantity field must be equal to the total quantity.	Char
38	OrderQty	Y	Quantity ordered. This value represents the number of shares for equities or par, face or nominal value for Fixed Income instruments.	Qty
40	OrdType	Y	Indicates the type of order. Valid values are: 1 – Market 2 – Limit 3 – Stop/Stop Loss 4 – Stop Limit Z – Market at Best	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
44	Price	Y/N	Required for all limit order types – not required for Market orders, or yield based products. For yield based products, this is the clean price and does not include accrued interest. Price (44) and Yield (236) are mutually exclusive.	Price
54	Side	Y	Side of the market. Valid values are: 1 – Buy 2 – Sell 5 – Regulated Short Sell(RSS) 6 – Proprietary Day Trading(PDT) I - Intraday Short Sell(IDSS) V – Permitted Short Sell(PSS)	Char
60	TransactTime	Y	Time of order creation by Trader. This field is not processed by the Exchange nor is it used as a mechanism to place an order at a future time.	UTCTimeStamp
110	MinQty	Y/N	Specifies the minimum fill quantity. Required if an All or None quantity condition and must be equal to the total quantity.	Qty
1138	DisplayQty	Y/N	Specifies the disclosed volume on hidden/iceberg orders.	Qty
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Absence of this field indicates a 'day' order.	Char
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
528	OrderCapacity	N	Designates the capacity of the firm placing the order. Valid values are: A – Agency P – Principal M – Market Maker R – Riskless Principal	Char
529	OrderRestrictions	Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
58	Text	N	Free Text.	String (24)
Standard Trailer		Y		

3.10 New Order Cross (s)

The New Order Cross type is used to submit a cross order into a market. The cross order contains two order sides (a buy and a sell). The cross order is identified by its CrossID.

Table 12 – New Order Cross

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = s	
548	CrossID		Y	Identifier for a cross order. Must be unique during a given trading day. Maximum length 20 characters.	String (20)
549	CrossType		Y	Type of cross being submitted to a market. Must be 1.	Int
550	CrossPrioritization		Y	Indicates if one side or the other of a cross order should be prioritized. Must be 0.	Int
Start of Component block, expanded in line < SideCrossOrdModGrp >					
552	NoSides		Y	Must be 2	NumInGrp
→	54	Side	Y	Side of order	Char
→	11	ClOrdID	Y	Unique identifier of the order as assigned by institution or by the intermediary with closest association with the investor.	String (20)
→	Component block <Parties>		Y N	Insert here the set of "Parties" fields. PartyRole 11 (Trader or Dealer ID) PartyRole 3 (Client ID, free to use) See Appendix B.	String (20) String (24)
→	1	Account	Y	Specifies Investor Account field. This is the 9-digit CDS account. Left-padded with "0" when required. E.g. "000181818".	String (9)
→	38	OrderQty	Y	Quantity ordered. This value represents the number of shares for equities or par, face or nominal value for Fixed Income instruments.	Qty
→	528	OrderCapacity	N	Designates the capacity of the firm placing the order	Char
→	529	OrderRestrictions	Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
→	58	Text	N	Free format text string	String (24)
End of Component block, expanded in line < SideCrossOrdModGrp >					
Start of Component block, expanded in line <Instrument>					
48	SecurityID		Y	Unique marketplace assigned identifier number for an order book. Eg. "1818", "1818WA".	String
22	SecurityIDSource		Y	Valid values: 99 – Marketplace assigned identifier	String
762	SecuritySubType		Y	Specifies the board on which SecurityID is listed. Valid value is: NM – Normal board	String
End of Component block, expanded in line <Instrument>					

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
40	OrdType	Y	Indicates the type of order. Valid value is: 2 – Limit	Char
44	Price	Y	Required for all limit order types.	Price
60	TransactTime	Y	Time of order creation by Trader. This field is not processed by the Exchange nor is it used as a mechanism to place an order at a future time.	UTCTimeStamp
59	TimeInForce	Y	Valid value: 3 – Immediate or Cancel (IOC)	Char
Standard Trailer		Y		

3.11 Order Cancel Request (F)

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

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

The format of the cancel request message is:

Table 13 – Order Cancel Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = F	
11	ClOrdID	Y	Unique identifier for Order as assigned by firms. See Appendix C. Maximum length 20 characters.	String (20)
37	OrderID	N	Unique order identifier as assigned by BTS2 that identifies the Order to be changed. Maximum length 18 characters.	String (18)
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID. Maximum length 20 characters.	String (20)
54	Side	Y	Side of the market.	Char
60	TransactTime	Y	Time this order request was initiated. This field is not processed by the Exchange nor is it used as a mechanism to cancel an order at a future time.	UTCTimeStamp
Standard Trailer		Y		

3.12 Order Cancel/Replace Request (G)

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

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

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

Table 14 – Order Cancel/Replace Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = G	
11	ClOrdID	Y	Unique identifier for Order as assigned by firms. See Appendix C. Maximum length 20 characters.	String (20)
583	ClOrdLinkID	N	Unique identifier to link One Cancel Other(OCO). Valid value: ClOrdID(11) of the previous non-rejected order.	String (20)
37	OrderID	N	Unique identifier of most recent order as assigned by the Exchange. Maximum length 18 characters.	String (18)
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID. Maximum length 20 characters.	String (20)
Component block <Parties>		Y N	Insert here the set of "Parties" fields. PartyRole 11 (Trader or Dealer ID) PartyRole 3 (Client ID, free to use) See Appendix B.	String (20) String (24)
Start of Component block, expanded in line <Instrument>				
48	SecurityID	Y	Unique marketplace assigned identifier number for an order book. Eg. "1818", "1818WA".	String
22	SecurityIDSource	Y	Valid value is: 99 – Marketplace assigned identifier	String
762	SecuritySubType	Y	In BTS2, this field is used to specify board on which SecurityID is listed. Valid values are: NM – Normal board OD – Odd-lot board BI – Buy-IN board	String
End of Component block, expanded in line <Instrument>				
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields.	
1	Account	Y	Specifies Investor Account field. This is the 9-digit CDS account. MUST MATCH THE VALUE OF THE LAST EXECUTION REPORT.	String (9)
18	ExecInst	N	Instructions for order handling. Only 'G' option is allowed (All or None). Note, whenever using this value to specify an All or None order the	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Minimum Quantity field (if provided) must be equal to the total quantity.	
38	OrderQty	Y	Quantity ordered.	Qty
40	OrdType	Y	Indicates the type of order to change to (must follow rules of the Exchange).	Char
44	Price	Y/N	Required for all limit order types, but not for yield based products. For yield based products, this is the clean price and does not include accrued interest. Price (44) and Yield (236) are mutually exclusive.	Price
54	Side	Y	Side of the market.	Char
60	TransactTime	Y	Time of execution/order creation. This field is not processed by the Exchange nor is it used as a mechanism to amend an order at a future time.	UTCTimeStamp
110	MinQty	N	Specifies the minimum fill quantity.	Qty
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment.	Char
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
1138	DisplayQty	Y/N	Specifies the disclosed volume on hidden/iceberg orders.	Qty
529	OrderRestrictions	Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
58	Text	N	Free Text.	String (24)
Standard Trailer		Y		

3.13 Order Cancel Reject (9)

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

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

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

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

The order cancel reject message format is as follows.

Table 15 – Order Cancel Reject

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = 9	
11	ClOrdID	Y	Unique identifier for Order as assigned by sell-side (e.g. exchange, ECN). If CxlRejReason="Unknown order" specify "NONE".	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
37	OrderID	Y	Unique identifier of most recent order as assigned by the Exchange. If CxlRejReason="Unknown order", specify "NONE".	String
39	OrdStatus	Y	Describes the current status of the order	Char
41	OrigClOrdID	Y/N	ClOrdID(11) of the previous non-rejected order (NOT the initial order of the day) when cancelling or replacing an order. Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID.	String
60	TransactTime	Y	Time of order cancellation request rejection by the Exchange.	UTCTimeStamp
102	CxlRejReason	Y	Code to identify reason for cancel rejection. Only '99' (Other) will be returned. Refer to 'text' (58) for exact reason for rejection.	Int
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to.	Char
58	Text	N	Specify BTS2 generated error message.	String
Standard Trailer		Y		

3.14 Order Status Request (H)

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

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

Table 16 – Order Status Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = H	
11	ClOrdID	Y	Corresponds to the ClOrdID (11) of the order whose status is being requested (if it exists). Conditionally required if the OrderID(37) is not provided. Either OrderID or ClOrdID must be provided. Maximum length 20 characters.	String (20)
37	OrderID	N	Conditionally required if ClOrdID(11) is not provided. Either OrderID (37) or ClOrdID (11) must be provided. Maximum length 18 characters.	String (18)
790	OrdStatusReqID	N	Optional, can be used to uniquely identify a specific Order Status Request message. Echoed back on Execution Report if provided. Maximum length 20 characters.	String (20)
54	Side	Y	Side of the market. This field is not processed by the Exchange.	Char
StandardTrailer		Y		

3.15 Order Mass Action Request (CA)

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

Table 17 - Order Mass Action Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = CA	
11	ClOrdID	Y	Unique ID of Order Mass Action Request as assigned by the institution. Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day.	String (20)
526	SecondaryClOrdID	N	Assigned by order originator. Maximum length 20 characters.	String (20)
584	MassStatusReqID	Y	Value assigned by issuer of Order Mass Action Request to uniquely identify the request. This ID will be returned on the Execution report. Maximum length 20 characters.	String (20)
1373	MassActionType	Y	Specifies the type of action requested. Valid values are: 3 – Cancel orders 100 – Order Status	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. Valid values: 1 - All orders for a security 7 - All orders 9 - All orders for a Market Segment NOTE: 7 and 9 are not supported when MassActionType=3	Int
1300	MarketSegmentID	N	Specifies the board on which SecurityID is listed. This field is equivelant to MarketSegmentID (1300). Valid values are: NM – Normal board OD – Odd-lot board BI – Buy-IN board	String
Component block <Instrument>		N	Insert here the set of "Instrument" (symbology) fields.	
54	Side	N	Side of the market.	Char
60	TransactTime	Y	Time of mass order action request by Trader. This field is not processed by the Exchange nor is it used to schedule an action at a future time.	UTCTimeStamp
Standard Trailer		Y		

3.16 Order Mass Action Report (BZ)

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

Table 18 - Order Mass Action Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = BZ	
11	ClOrdID	N	ClOrdID provided on the Order Mass Action Request.	String
526	SecondaryClOrdID	N	Assigned by order originator.	String
1369	MassActionReportID	Y	Unique Identifier for the Order Mass Action Report. This is a V5.0 tag value.	String
1373	MassActionType	Y	Specifies the type of mass action requested. This is a V5.0 tag value.	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. This is a V5.0 tag value.	Int
1375	MassActionResponse	Y	Indicates the action taken by the counterparty order handling system as a result of the Action Request. This is a V5.0 tag value.	Int
1376	MassActionRejectReason	N	Indicates why Order Mass Action Request was rejected. Required if MassActionResponse = 0 Reason Order Mass Action Request was rejected. This is a V5.0 tag value.	Int
533	TotalAffectedOrders	N	Optional field used to indicate the total number of orders affected by the Order Mass Action Request	Int
Start of Component block, expanded in line < AffectedOrdGrp >				
534	NoAffectedOrders	N	Optional field used to indicate the number of order identifiers for orders affected by the Order Mass Action Request. Must be followed with OrigClOrdID as the next field	Int
→	41 OrigClOrdID	N	Required if NoAffectedOrders > 0 and must be the first repeating field in the group. Indicates the client order id of an order affected by this request. If order(s) were manually delivered (or otherwise not delivered over FIX and not assigned a ClOrdID) this field should contain string "MANUAL".	String
End of Component block, expanded in line < AffectedOrdGrp >				
1300	MarketSegmentID	N	Market Segment where the security trades. It is mapped to BTS2 Board Id. This is a V5.0 tag value.	String
Component block <Instrument>		N	Insert here the set of "Instrument" (symbology) fields.	
54	Side	N	Side of the market.	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
60	TransactTime	N	Equal to time of Order Mass Action Request.	UTCTimeStamp
58	Text	N	Free format text string	String
Standard Trailer		Y		

3.17 Execution Report (8)

The execution report message is used to:

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

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

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

Table 19 – Execution Report

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = 8	
11	ClOrdID	Y/N	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Required when referring to orders that where electronically submitted over FIX or otherwise assigned a ClOrdID(11). In the case of quotes can be mapped to the QuoteID (117) tag of a Mass Quote.	String
17	ExecID	Y	Unique identifier of execution message as assigned by the Exchange (will be 0 (zero) for ExecType=I (Order Status)).	String
18	ExecInst	N	Instructions for order handling. Only 'G' option can be specified (All or None).	Char
37	OrderID	Y	OrderID is required to be unique for each chain of orders.	String
198	SecondaryOrderID	N	The original OrderID that is replaced by the the	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			current OrderID. It might be the same as the OrderID. Only available after an order is successfully replaced.	
41	OrigClOrdID	Y/N	Conditionally required for response to a Cancel or Cancel/Replace request	String
583	ClOrdLinkID	Y/N	Unique identifier to link One Cancel Other(OCO). Valid value: ClOrdID(11) of the previous non-rejected order.	String
150	ExecType	Y	Type of Execution being reported. Describes the specific ExecutionRpt (i.e. Pending Cancel) while OrdStatus (39) will always identify the current order status (i.e. Partially Filled).	Char
526	SecondaryClOrdID	N	Assigned by the party that originates the order. In the case of quotes can be mapped to QuoteEntryID (299) of a Mass Quote.	String
584	MassStatusReqID	Y/N	Required if responding to an Order Mass Status Request. Echos back the value provided by the requester.	String
790	OrdStatusReqID	N	Required if responding to and if provided on the Order Status Request message. Echo back the value provided by the requester.	String
911	TotNumReports	N	Can be used when responding to an Order Mass Status Request to identify the total number of Execution Reports which will be returned. Not Supported.	Int
961	HostCrossID	N	Host assigned entity ID that can be used to reference all components of a cross	String
548	CrossID	N	Identifier for a cross order.	String
Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields.	
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields.	
Component block <TriggeringInstruction>		N	Insert here the set of "TriggeringInstruction" fields.	
1	Account	Y	Specifies Investor Account field. This is the 9-digit CDS account. Left-padded with "0" when required, eg. "000181818".	String
6	AvgPx	N	Calculated average price for all fills on this order during the day. If not available then the value reflects the trade price for this fill.	Price
14	CumQty	Y	Total matched quantity.	Qty
31	LastPx	N	Price of this fill.	Price
32	LastQty	N	Quantity (e.g. shares) bought/sold on this fill.	Qty
38	OrderQty	N	Quantity ordered.	Qty
110	MinQty	N	Minimum fill quantity.	Qty

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
39	OrdStatus	Y	Describes the current state of an order.	Char
40	OrdType	N	OrderType	Char
44	Price	N	Price on order.	Price
54	Side	Y	Side of order.	Char
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Absence of this field indicates a 'day' order.	Char
60	TransactTime	Y	Time of execution/order creation (expressed in Universal Time Coordinated (UTC), also known as GMT.	UTCTimeStamp
75	TradeDate	N	Indicates date of trade referenced in this message in YYYYMMDD format.	LocalMktDate
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.	LocalMktDate
126	ExpireTime	Y/N	Conditionally required if TimeInForce = GTD and ExpireDate is not specified.	UTCTimestamp
64	SettlDate	N	Specific date of trade settlement Settlement Date is in YYYYMMDD format.	LocalMktDate
103	OrdRejReason	N	For optional use with ExecType = 8 (Rejected). Code to identify reason for order rejection.	Int
378	ExecRestatementReason	N	Code to identify reason for order cancel or expire.	Int
151	LeavesQty	Y	Quantities open for further execution. If the OrdStatus is Cancelled, DoneForTheDay, Expired or Rejected (in which case the order is no longer active) then LeavesQty could be 0, otherwise LeavesQty = OrderQty – CumQty.	Qty
236	Yield	N	Yield percentage (Fixed Income only)	Percentage
381	GrossTradeAmt	N	Total amount traded expressed in units of currency. Includes accrued Interest for convertible bonds and fixed income.	Amt
159	AccruedInterestAmt	N	Amount of Accrued Interest for convertible bonds and fixed income	Amt
880	TradeMatchID	N	Identifier assigned by the trading system for a trade. This is the TRS number.	String
1057	AggressorIndicator	N	Used to identify whether the order initiator is an aggressor or not in the trade. Valid during continuous trading only. This is a V5.0 tag value.	Boolean
1138	DisplayQty	N	Replaces 'MaxFloor' and specifies the disclosed volume on hidden/iceberg orders. This is a V5.0 tag value. This field is always returned as part of a fill or partial fill for all order types. For non-hidden/iceberg orders this field will contain the same value as LeavesQty (151).	Qty
528	OrderCapacity	N	Designates the capacity of the firm placing the	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			order	
529	OrderRestrictions	N	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
58	Text	N	Free text. On an error condition, this will specify BTS2 generated error message.	String
797	CopyMsgIndicator	N	Drop Copy	Boolean
Standard Trailer		Y		

Table 20 – Execution Report Returned Tags Based On Scenario

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

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

R = Returned as part of Execution Report message

4 Quote Management

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

- Mass Quote
- Mass Quote Acknowledgement

4.1 Unique QuoteID (117)

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

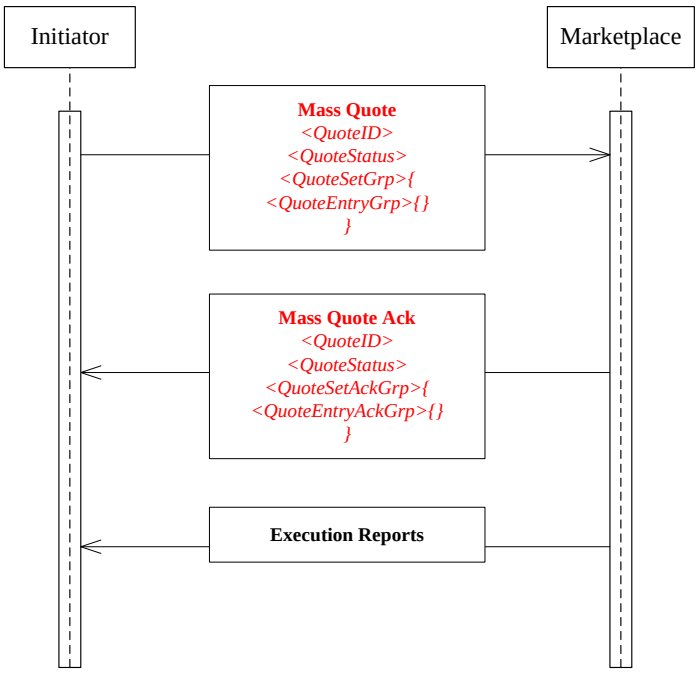
4.2 Workflows

4.2.1 Entering of a mass quote

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

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

Figure 10 – Mass Quote workflow



Execution Reports will be generated for quotes individually. The ClOrdID(11) field will be set to QuoteID(117), the SecondaryClOrdID(526) will be set to QuoteEntryID(299). Every quote will be assigned an unique OrderID(37). OrderCapacity (528) on the execution report will be set to 'M'- Market maker order.

4.2.2 Quote Entry Cancel

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

If a quote is cancelled by the system or expired, an ExceptionReport will be generated.

4.3 Mass Quote (i)

The Mass Quote message can contain quotes for multiple securities to support applications that allow for the mass quoting of an option series.

BTS2 FIX supports only one QuoteSetGrp per Mass Quote message. A QuoteSetGrp can have maximum 12 QuoteEntryGrps. Fragmented Mass Quote messages will be rejected - NoQuoteEntries(295) should always be the same as TotNoQuoteEntries(304) in the same QuotSetGrp.

Table 21 – Mass Quote

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = i	
117	QuoteID		Y	Unique identifier for quote	String
537	QuoteType		N	Type of Quote. Valid values: 100 – Permitted Short Sell(PSS) 101 – Permitted Short Sell and withdraw on log off 102 - Withdraw on log off	Int
Component block <Parties>			Y	PartyRole 11 (Trader or Dealer ID)	String (20)
1	Account		Y	Specifies trade account type.	String
529	OrderRestrictions		Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
Start of Component block, expanded in line < QuotSetGrp >					
296	NoQuoteSets		Y	The number of sets of quotes in the message	NumInGrp
→	302	QuoteSetID	Y	Sequential number for the Quote Set. For a given QuoteID – assumed to start at 1. Must be the first field in the repeating group.	String
→	304	TotNoQuoteEntries	Y	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set.	Int
→	Start of Component block, expanded in line < QuotEntryGrp >				
→	295	NoQuoteEntries	Y	The number of quotes for this QuotSet	NumInGrp

TAG	FIELDNAME			REQ'D	COMMENTS	FORMAT
					that follow in this message.	
→	→	299	QuoteEntryID	Y	Uniquely identifies the quote across the complete set of all quotes for a given quote provider.	String
→	→	Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields defined in "Common Components of Application Messages.	
→	→	132	BidPx	N	Bid price/rate.	Price
→	→	133	OfferPx	N	Offer price/rate.	Price
→	→	134	BidSize	N	Quantity of bid	Qty
→	→	135	OfferSize	N	Quantity of offer	Qty
→	End of Component block, expanded in line < QuotEntryGrp >					
End of Component block, expanded in line < QuotSetGrp >						
Standard Trailer				Y		

4.4 Mass Quote Acknowledgement (b)

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

Table 22 – Mass Quote Acknowledgement

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = b	
117	QuoteID	Y	Unique identifier for quote	String
537	QuoteType	N	Type of Quote. Valid values: 100 – Permitted Short Sell(PSS) 101 – Permitted Short Sell and withdraw on log off 102 - Withdraw on log off	Int
297	QuoteStatus	N	Status of the mass quote acknowledgement.	Int
300	QuoteRejectReason	N	Reason Quote was rejected.	Int
Component block <Parties>		Y	PartyRole 11 (Trader or Dealer ID)	String (20)
1	Account	Y	Specifies trade account type.	String
529	OrderRestrictions	Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultiCharValue
58	Text	N	Free format text string	String
Start of Component block, expanded in line < QuotSetAckGrp >				
296	NoQuoteSets	Y	The number of sets of quotes	NumInGrp

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				in the message	
→	302	QuoteSetID	Y	Sequential number for the Quote Set. For a given QuoteID - assumed to start at 1. Must be the first field in the repeating group.	String
→	304	TotNoQuoteEntries	N	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set. Required if NoQuoteEntries > 0	Int
→	893	LastFragment	N	Indicates whether this is the last fragment in a sequence of message fragments. Only required where message has been fragmented.	Boolean
→	Start of Component block, expanded in line < QuotEntryAckGrp >				
→	295	NoQuoteEntries	Y	The number of quotes for this QuotSetAck that follow in this message.	NumInGrp
→	→	299 QuoteEntryID	Y	Uniquely identifies the quote across the complete set of all quotes for a given quote provider.	String
→	→	Component block <Instrument>	Y	Instrument component received in QuotEntryGrp	
→	→	132 BidPx	N	Bid price/rate.	Price
→	→	133 OfferPx	N	Offer price/rate.	Price
→	→	134 BidSize	N	Quantity of bid	Qty
→	→	135 OfferSize	N	Quantity of offer	Qty
→	→	62 ValidUtilTime	N	Indicates expiration time of quote (always expressed in UTC (Universal Time Coordinated, also known as "GMT")	UTCTimestamp
→	→	1167 QuoteEntryStatus	N	Identifies the status of an individual quote.	Int
→	→	368 QuoteEntryRejectReason	N	Reason Quote Entry was rejected.	Int
→	End of Component block, expanded in line < QuotEntryAckGrp >				
End of Component block, expanded in line < QuotSetAckGrp >					
Standard Trailer			Y		

5 Account Modification

The Allocation Instruction message is used for change the account of a filled or partially filled order, or reported trade, or a quote in a mass quote. The Allocation Instruction Ack message is used to acknowledge the receipt of and provide status for an Allocation Instruction message.

5.1 Allocation Instruction (J)

To change the account specified in a filled or partially filled order, or reported trade, use Allocation Instruction message. If the message is accepted, each affected order and/or trade will generate an Execution Report message to reflect the current status. An Allocation Instruction Ack message will always be returned regardless of the result.

To change the account to same account[CDS account] then PartyRole(3) with different client ID must provided else the allocation instruction will rejected.

Table 23 – Allocation Instruction

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
StandardHeader			Y	MsgType = J	
70	AllocID		Y	Unique identifier for this allocation instruction message.	String
71	AllocTransType		Y	Always 1 = Replace	char
626	AllocType		Y	Specifies the purpose or type of Allocation message. Always 9 = Accept.	Int
Start of Component block, expanded in line < OrdAllocGrp >					
73	NoOrders		N	Indicates number of orders to be modified. It should always be 1	Int
→	11	ClOrdID	Y	ClOrdID of the order to be amended. To amend an order that doesn't have ClOrdID, or a quote in mass quote, this field should contain string "MANUAL". Maximum length 20 characters.	String
→	37	OrderID	N	BTS2 OrderID of the order to be amended. It can be the order ID of a quote in a mass quote, or the buy or sell side OrderID of a reported trade as well. Maximum length 18 characters.	String
End of Component block, expanded in line < OrdAllocGrp >					
54	Side		Y	Side of the order to be amended. It will not be validated.	char
Component block <Instrument>			Y	Insert here the set of "Instrument" (symbology) fields of the original order. They will not be validated.	
53	Quantity		Y	Set to 0.	Qty
Component block <Parties>			N	Insert here the set of "Parties" fields. PartyRole 3 (Client ID, free to use)	String (24)
75	TradeDate		Y	The current market date. It's not validated.	LocalMktDate
Start of Component block, expanded in line < AllocGrp >					
78	NoAllocs		Y	Indicates number of accounts to be modified. It	Int

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
				should always be 1.	
→	79	AllocAccount	Y	The new account of the order. This is the 9-digit CDS account. Left-padded with "0" when required, eg. "000181818".	String (9)
End of Component block, expanded in line < AllocGrp >					
StandardTrailer			Y		

5.2 Allocation Instruction Ack (P)

The Allocation Instruction Ack message is used to acknowledge the receipt of and provide status for an Allocation Instruction message.

Table 24 – Allocation InstructionAck

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = P	
70	AllocID	Y	The AllocID in Allocation Instruction message	Char
87	AllocStatus	Y	Identifies status of allocation. Valid values: 0 – accepted 1 – reject	Int
88	AllocRejCode	N	Required for AllocStatus = 1. Valid values: 0 – Unknown account 5 – Unknown OrderID 12 – Unknown ClOrdID 99 – Other	Int
58	Text	N	Reason for reject	String
StandardTrailer		Y		

6 Trade Capture Reporting

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

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

6.1 Trade Capture Messages

The Trade Capture category of messages consists of the following:

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

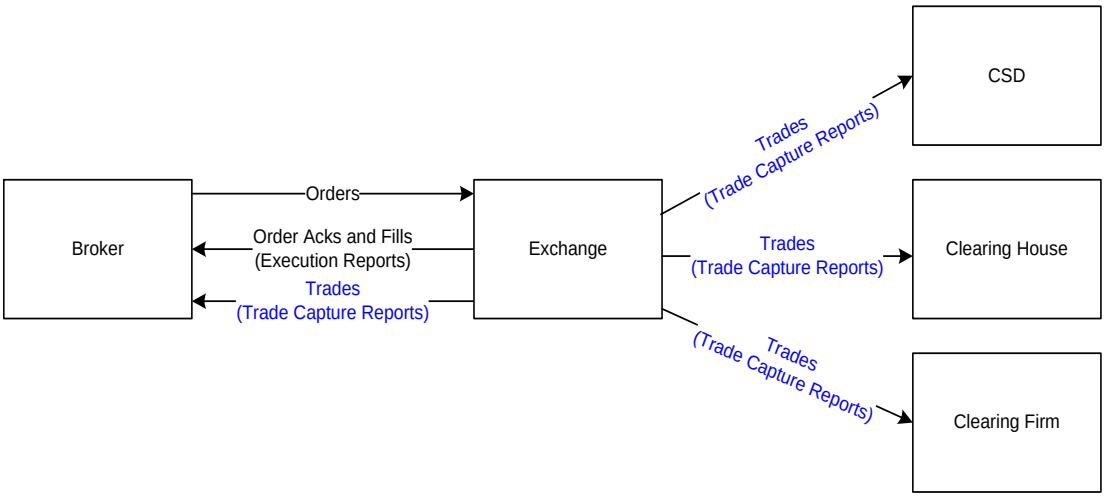
6.2 Workflows

Trade Capture Reports are used for various purposes including:

- Relaying Confirmed Trades to various parties not directly involved in the execution, e.g. CSDs, clearing houses, clearing firms and regulatory bodies. Those messages are **outbound** (from the marketplace).
- Relaying Confirmed Trades to counterparties of the trade. Where Execution Reports may be sufficient for front-office purposes, Trade Capture Reports can serve more demanding back-office processes better. Those messages are **outbound** (from the marketplace).
- Reporting of privately negotiated ("off-market") trades, i.e. trades formed outside of the marketplace. Those messages are **inbound** (to the marketplace) but may also be used as **outbound** (when the marketplace relays them to counterparties). Seller always report two sided negotiated trades.

6.2.1 Trade Capture Workflow for Multiple Counterparties

Figure 11 – Trade Capture High Level Workflow

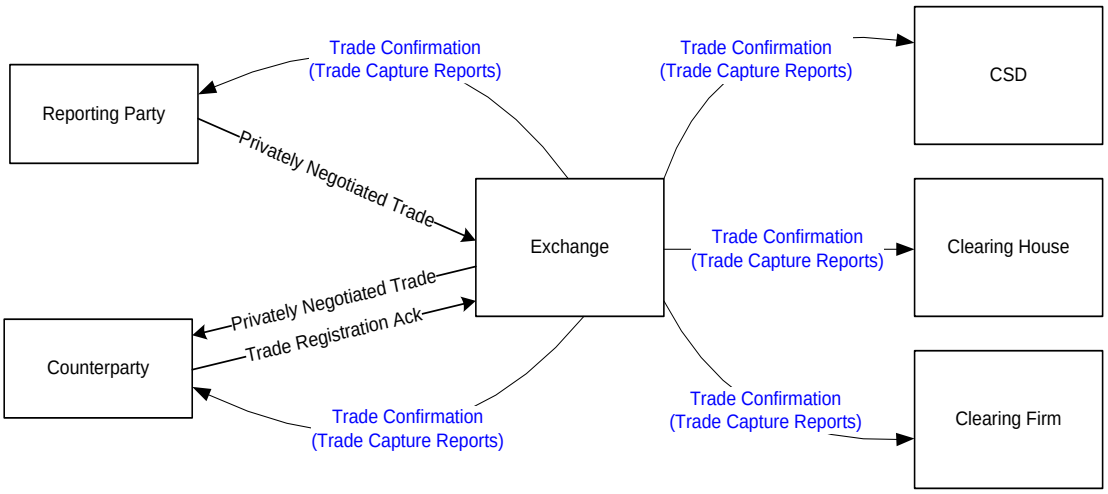


6.2.2 Trade Capture Diagram for Privately Negotiated Trade, One-Party Report for Pass-through to Counterparty

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

When submitting a one sided (crossing) Trade Capture Report, the submitter must fill in details of both sides in two TrdCapRptSideGrps. The Exchange will either confirm the TradeCaptureReport, or reject with TradeCaptureReportAck.

Figure 12 – Diagram for Negotiated Trade



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

1. The initiator (seller) sends TradeCaptureReport (AE) with a unique TradeReportID (571).
2. If rejected, the marketplace will send to the initiator a TradeCaptureReport (AE) with the TradeReportID (571) set to the same as the TradeReportID (571) received.
3. If accepted, the marketplace will send to the initiator a TradeCaptureReport (AE) with a new TradeReportID (571), a new ExecID (17) and the TradeReportRefID (572) set to the initiator's TradeReportID (571). The marketplace will also send a TradeCaptureReport (AE) to the counterparty, with a new TradeReportID (571) and a new ExecID (17).
4. The initiator can withdraw the TradeCaptureReport with a new TradeReportID (571), a ExecID (17) that is set to the same ExecID in the TradeCaptureReport received from the marketplace in step 3. And tradeReportRefID (572) set to the same as the TradeReportID (571) received in step 3.
5. The counterparty can either accept or reject the alleged TradeCaptureReport with a new TradeCaptureReport, by setting the ExecID (17) to the same as the ExecID (17) in the TradeCaptureReport received from the marketplace. The TradeReportRefID (572) should be set to the same as the TradeReportID (571) in the TradeCaptureReport received from the Marketplace.
6. If the counterparty confirms, the marketplace will send a TradeCaptureReport to both parties, with a new ExecID (17) and TradeReportRefID (572) set to the same as the TradeReportID (571) sent in step 3. The MatchStatus (573) will be set to 0 – affirmed.
7. If the counterparty declines or the initiator withdraws the trade, the marketplace will send a TradeCaptureReport with both ExecID (17) and tradeReportID (571) set to be the same as in step 3. The initiator will receive TradeReportRefID (572) set to the same as the TradeReportID (571) sent in step 1.

Figure 13 - One-Party Report for Pass-through to Counterparty

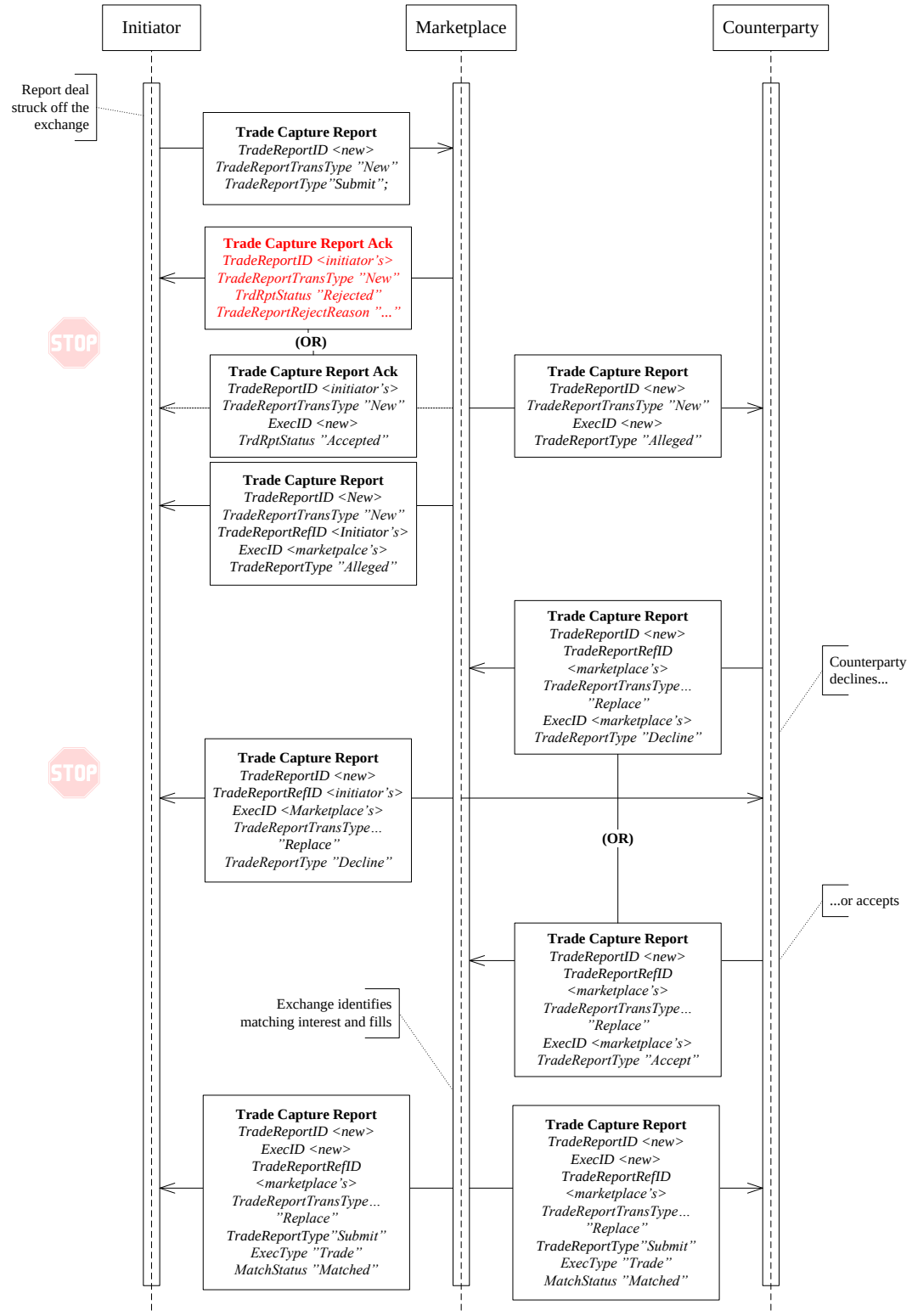
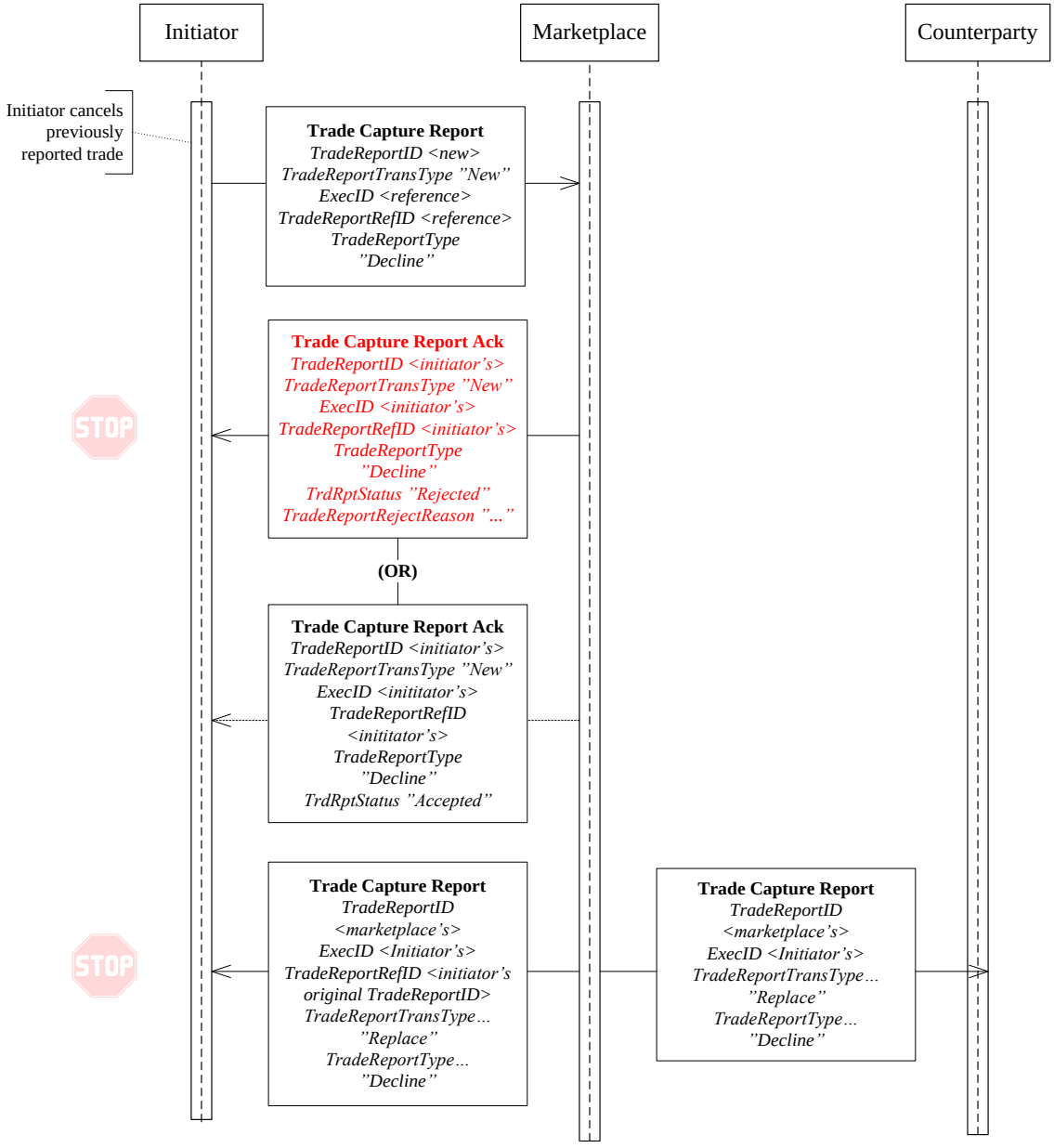
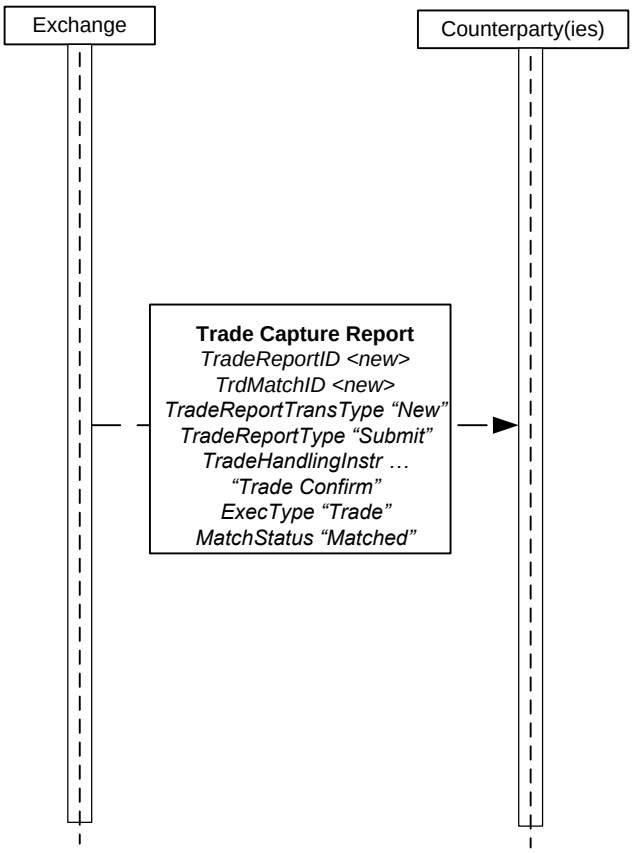


Figure 14 - One-Party cancel Report (or time out) before counterparty confirms



6.2.4 Workflow for Third Party Trade Capture

Figure 15 – Third Party Trade Capture



6.3 Trade Capture Report Request (AD)

The Trade Capture Report Request is used to:

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

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

Table 25 – Trade Capture Report Request

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AD	
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default (snapshot only – no subscription).	Char
568	TradeRequestID	Y	Identifier for the trade request. Maximum length 20 characters.	String
569	TradeRequestType	Y	Type of Trade Capture Report.	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardTrailer		Y		

6.4 Trade Capture Report (AE)

The Trade Capture Report message can be:

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

When reporting an off market two party trade, the seller[only seller allow to initiate the Trade Capture Report] should fill in two TrdCapRptSideGrp, one buy side and the other sell side. In the counter party side, the message should contain trader ID in PartyID (448) with PartyRole (452) = 12 (Executing Trader) in the Parties block.

Reporting an off market crossing trade doesn't require either trader ID or firm ID.

When the counterparty confirms a reported two party trade, only one TrdCapRptSideGrp is required(the counterparty side).

Table 26 – Trade Capture Report

TAG	FIELDNAME	REQ'D	COMMENTS	DIRECTION	FORMAT
StandardHeader		Y	MsgType = AE		
571	TradeReportID	Y	Unique identifier for the Trade Capture Report. Maximum length 20 characters.	In/Out	String
17	ExecID	N	Exchanged assigned Execution ID (Trade Identifier)	In/Out	String
487	TradeReportTransType	N	Identifies Trade Report message transaction type 0 – New 1 – Cancel 2 – Replace	In/Out	Int
856	TradeReportType	N	Type of Trade Report 0 – Submit 1 – Alleged 2 – Accept 3 – Decline 6 – Trade Report Cancel 10 – Pended	In/Out	Int
828	TrdType	N	Type of Trade. 0 – Regular Trade 22 – Privately Negotiated Trades 100 – Crossing Order Trade	Out	Int
150	ExecType	N	Type of Execution being reported.	Out	Char
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default.	Out	Char
572	TradeReportRefID	N	The TradeReportID that is being referenced for some action, such as correction or cancellation. Maximum	In/Out	String

TAG	FIELDNAME	REQ'D	COMMENTS	DIRECTION	FORMAT
			length 20 characters.		
570	PreviouslyReported	N	Indicates if the trade capture report was previously reported to the counterparty.	Out	Boolean
939	TrdRptStatus	N	Status of Trade Report 0 = Accepted 1 = Rejected	Out	Int
325	UnsolicitedIndicator	N	Set to 'Y' if message is sent as a result of a subscription request or out of band configuration as opposed to a Position Request.	Out	Boolean
568	TradeRequestID	N	Request ID if the Trade Capture Report is in response to a Trade Capture Report Request	Out	String
573	MatchStatus	N	The status of this trade with respect to matching or comparison 0 = compared, matched or affirmed 1 = uncompared, unmatched, or unaffirmed	Out	Char
574	MatchType	N	The point in the matching process at which this trade was matched.	Out	String
Component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields	In/Out	
Component block <YieldData>		N	Insert here the set of "YieldData" fields	In/Out	
15	Currency	N	Primary currency of the specified currency pair. Used to qualify LastQty and GrossTradeAmount	Out	Currency
31	LastPx	Y	Trade Price. Dirty price for yield traded bonds.	In/Out	Price
32	LastQty	Y	Trade Quantity	In/Out	Qty
60	TransactTime	N	Time the transaction represented by this Trade Capture Report occurred	Out	UTCTimeStamp
64	SettlDate	N	Specific date of trade settlement (Settlement Date) in YYYYMMDD format	In/Out	LocalMktDate
75	TradeDate	N	Used when reporting other than current day trades	Out	LocalMktDate
381	GrossTradeAmt	N	Total amount traded expressed in units of currency. Includes accrued Interest for convertible bonds and fixed income.	Out	Amt
880	TradeMatchID	N	Identifier assigned by the trading system for a trade. Maximum length 21 characters.	In/Out	String
Start of Component block, expanded in line < TrdCapRptSideGrp >					
552	NoSides	Y	Number of sides	In/Out	Int

TAG	FIELDNAME		REQ'D	COMMENTS	DIRECTION	FORMAT
→	54	Side	Y	Side of order	In/Out	Char
→	37	OrderID	N	OrderID should be conditionally required when Trade Capture Report is used for back office processing	Out	String
→	11	ClOrdID	N	Required for executions against electronically submitted orders which were assigned an ID by the institution or intermediary. In the case of quotes can be mapped to: - QuoteID(117) of a Mass Quote	Out	String
→	Component block <Parties>		Y N	Insert here the set of "Parties" fields. PartyRole 11 (Trader or Dealer ID) PartyRole 3 (Client ID, free to use) See Appendix B.	In/Out	
→	1	Account	Y	Specifies Investor Account field. This is the 9-digit CDS account. Left-padded with "0" when required, eg. "000181818".	In/Out	String
→	528	OrderCapacity	N	Designates the capacity of the firm placing the order.	In/Out	char
→	529	OrderRestrictions	Y	For order tagging purpose. Maximum length 5 characters.	In/Out	MultipleCharValue
→	159	AccruedInterestAmt	N	Amount of Accrued Interest for convertible bonds and fixed income	Out	Amt
End of Component block, expanded in line < TrdCapRptSideGrp >						
797	CopyMsgIndicator		N	Indicates Drop Copy	Out	Boolean
StandardTrailer			Y			

6.5 Trade Capture Report Ack (AR)

The Trade Capture Report Ack message can be:

- Used to acknowledge trade capture reports received from a counterparty
- Used to reject a trade capture report received from a counterparty

Table 27 – Trade Capture Report Ack

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AR	
571	TradeReportID	Y	Unique identifier for the Trade Capture Report	String
487	TradeReportTransType	N	Identifies Trade Report message transaction type 0 – New 1 – Cancel 2 – Replace	Int
856	TradeReportType	N	0 – Submit 1 – Alleged 2 – Accept	

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			3 – Decline 6 – Trade Report Cancel 10 – Pended	
150	ExecType	N	Type of Execution being reported	Int
939	TrdRptStatus	N	0 – Accepted 1 – Rejected	Int
17	ExecID	N	Exchanged assigned Execution ID (Trade Identifier)	String
751	TradeReportRejectReason	N	Reason for Rejection of Trade Report	Int
572	TradeReportRefID	N	The TradeReportID that is being referenced for some action, such as correction or cancellation	String
58	Text	N	If TradeReportRejectReason is set, text of reason	String
StandardTrailer		Y		

6.6 Trade Capture Report Request Ack (AQ)

The Trade Capture Report Request Ack message can be:

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

Table 28 – Trade Capture Report Request Ack

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
StandardHeader		Y	MsgType = AQ	
263	SubscriptionRequestType	N	Used to subscribe / unsubscribe for trade capture reports. If the field is absent, the value 0 will be the default.	Char
568	TradeRequestID	Y	Identifier for the trade request	String
569	TradeRequestType	Y	Type of Trade Capture Report.	Int
748	TotNumTradeReports	N	Number of trade reports returned	Int
749	TradeRequestResult	Y	Result of Trade Request.	Int
750	TradeRequestStatus	Y	Status of Trade Request.	Int
component block <Instrument>		Y	Insert here the set of "Instrument" (symbology) fields	
58	Text	N	Free format text string	String
StandardTrailer		Y		

6.7 BTS2 Trade Capture Report Scenario

- Inter-broker : Dealer A and B from two different Participating Organisations.
- Intra-broker : Dealer A and B from same Participating Organisations but tow different system/FIX connections
- Intra-broker : Dealer A and B from same Participating Organisations

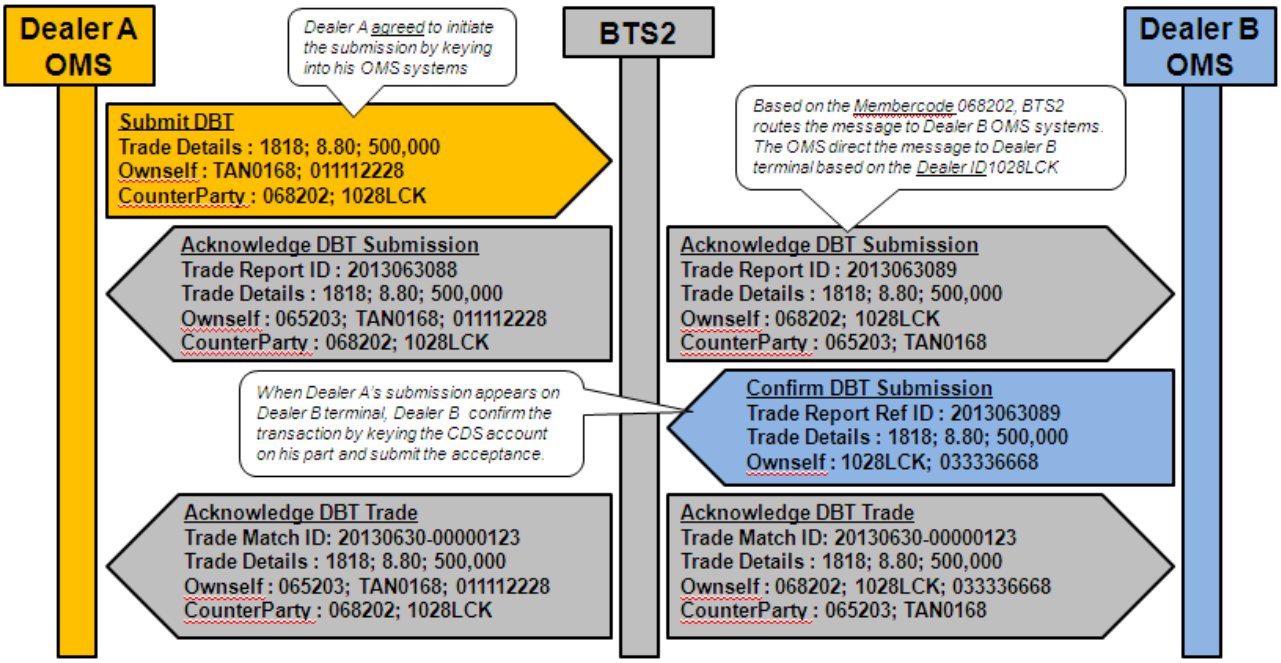
Important Details Prior to Submission of DBT

	Description
Trade Details	• Securities • Agreed Price & Quantity • CDS Account(s)
Systems Membercode	Every systems capable of submitting DBTs must have a Systems Membercode (eg. 065201, 098203 and 035202). BTS2 trading engine requires Membercodes of both parties for the purpose of routing the trade messages. It is advisable to have the Systems Membercode displayed on the DBT entry screen.
Dealer ID	Upon receiving trade messages from the BTS2 trading engine, every OMS will rely on the Dealer ID to route the trade to the right dealer's screen.
Order Entry Sequence	1st Step : Seller to submit the DBT 2nd Step : Buyer to confirm

DBT Scenario #1 : Inter-Broker

	Description
Trade Details	• Securities : 1818 • Agreed Price & Quantity : RM8.80 for 500,000 shares • CDS Account (s) • Dealer A (Seller) : 011112228 • Dealer B (Buyer) : 033336668
Systems Membercode	• Dealer A (Seller) : 065203 • Dealer B (Buyer) : 068202
Dealer ID	• Dealer A (Seller) : TAN0168 • Dealer B (Buyer) : 1028LCK

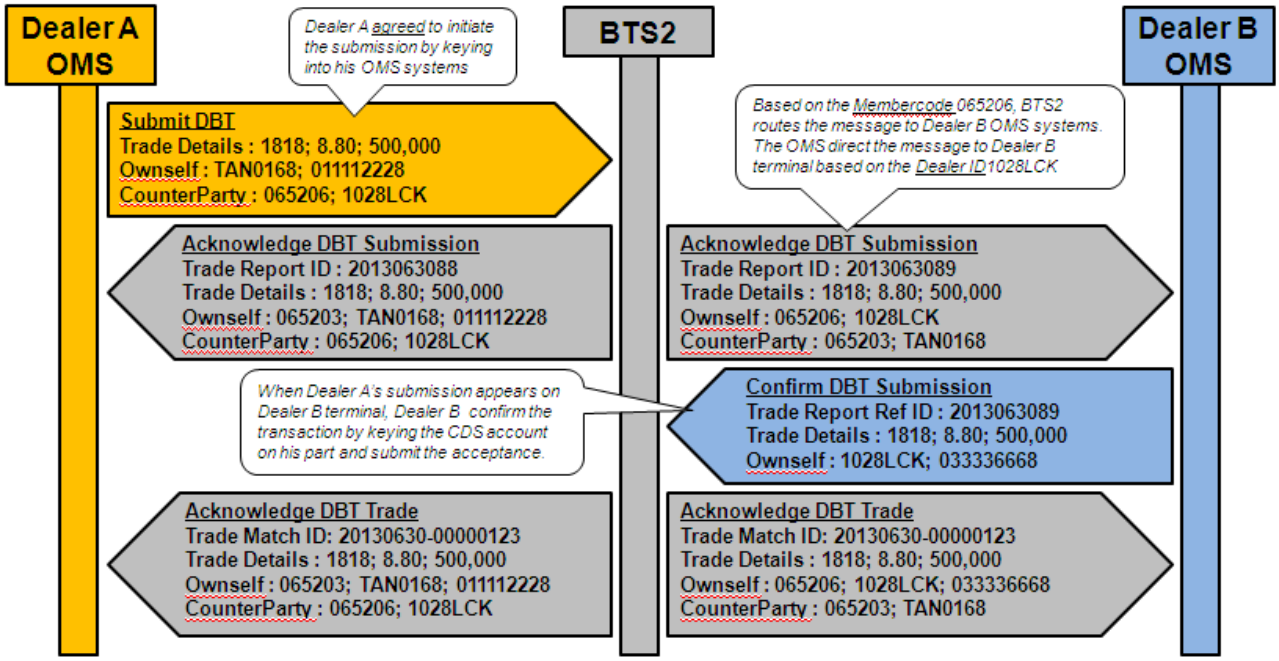
DBT Scenario #1 : Inter-Broker



DBT Scenario #2 : Intra-Broker, Different Systems

	Description
Trade Details	• Securities : 1818 • Agreed Price & Quantity : RM8.80 for 500,000 shares • CDS Account (s) • Dealer A (Seller) : 011112228 • Dealer B (Buyer) : 033336668
Systems Membercode	• Dealer A (Seller) : 065203 • Dealer B (Buyer) : 065206
Dealer ID	• Dealer A (Seller) : TAN0168 • Dealer B (Buyer) : 1028LCK

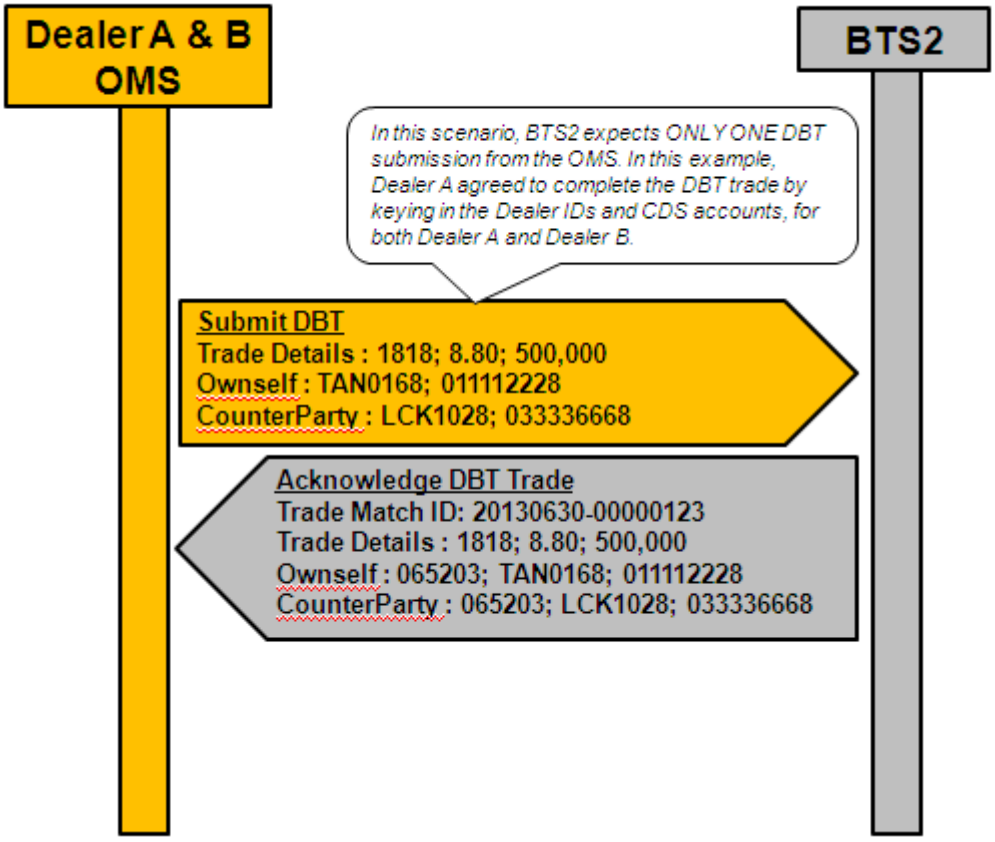
DBT Scenario #2 : Intra-Broker, Different Systems



DBT Scenario#3 : Intra-Broker, Same Systems

	Description
Trade Details	• Securities : 1818 • Agreed Price & Quantity : RM8.80 for 500,000 shares • CDS Account (s) • Dealer A (Seller) : 011112228 • Dealer B (Buyer) : 033336668
Systems Membercode	• Dealer A (Seller) : 065203 • Dealer B (Buyer) : 065203
Dealer ID	• Dealer A (Seller) : TAN0168 • Dealer B (Buyer) : LCK1028

DBT Scenario#3 : Intra-Broker, Same Systems



7 Trade Cancellation

Trade Capture reports are used to request and respond to requests for Trade Cancellation.

7.1 Workflows

The trade cancellation workflow consists of an initiating party submitting a trade cancellation request, and the counterparty accepting or declining.

Depending on BTS2 configuration, a trade cancellation affects immediately upon confirmation, or requires Market Controller's approval.

7.1.1 Workflow for Trade Cancellation

Figure 16 – Trade Cancellation for Initiator

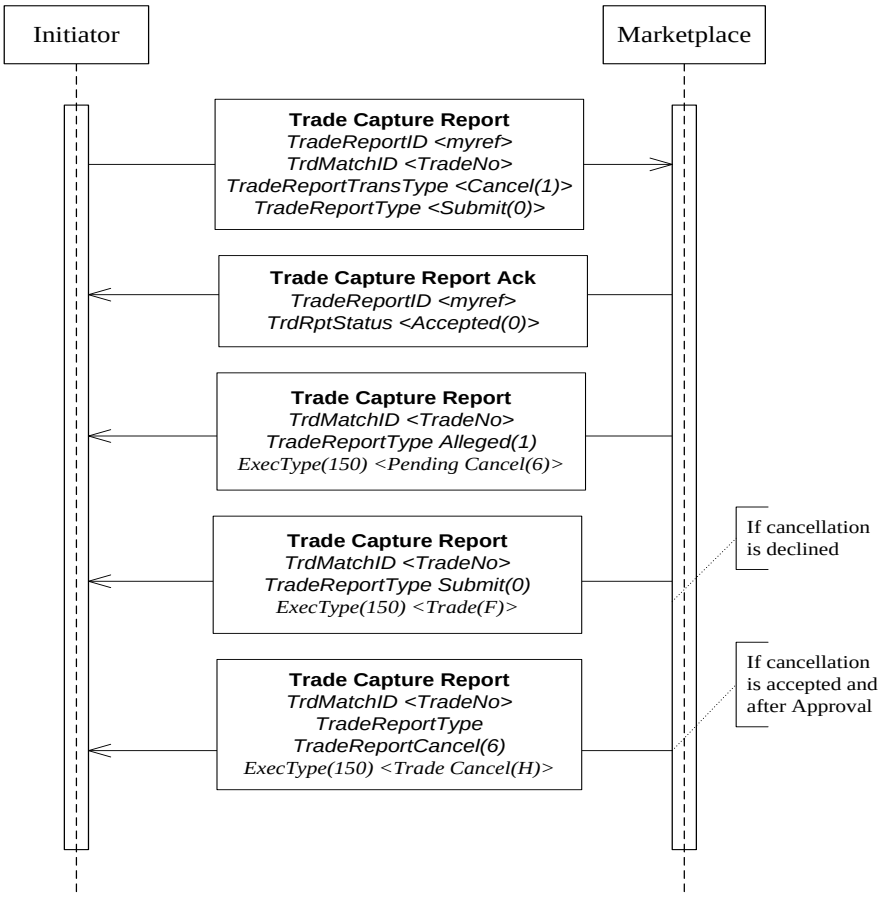


Figure 17 – Trade Cancel Reject for Counterparty

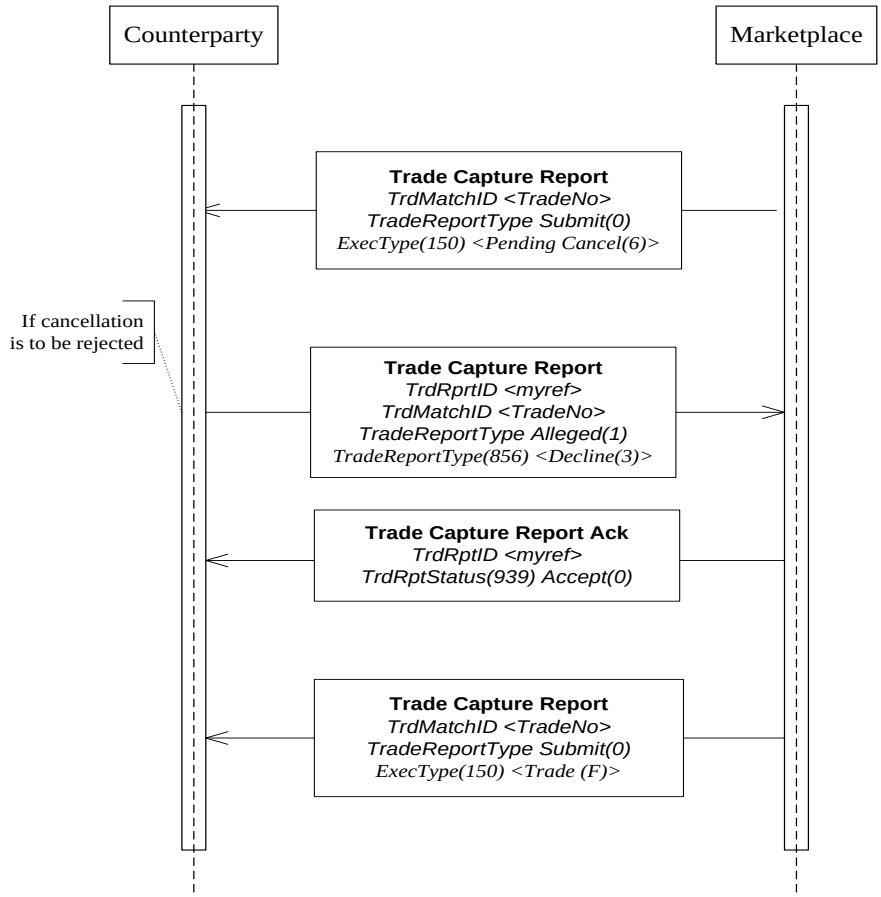
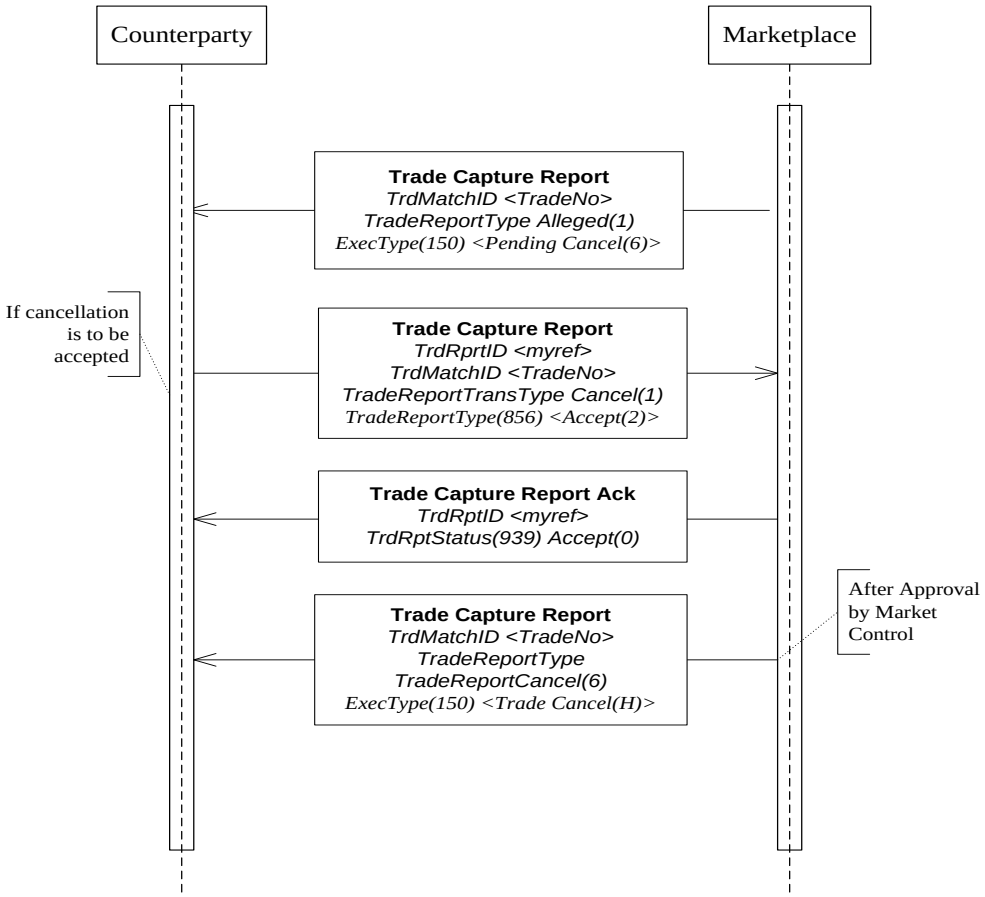


Figure 18 – Trade Cancel Accept for Counterparty



8 Market & Reference Data

The market & reference data category consists of the following messages:

- **Trading Session Status Request**
  Refer to BTS2 FIX Specification – Market Data, Section 8.7
- **Trading Session Status**
  Refer to BTS2 FIX Specification – Market Data, Section 8.8
- **Security Status Request**
  Refer to BTS2 FIX Specification – Market Data, Section 8.9
- **Security Status**
  Refer to BTS2 FIX Specification – Market Data, Section 8.10
- **Market Definition Request**
  Refer to BTS2 FIX Specification – Market Data, Section 9.2
- **Market Definition**
  Refer to BTS2 FIX Specification – Market Data, Section 9.3
- **Trading Session List Request**
  Refer to BTS2 FIX Specification – Market Data, Section 9.4
- **Trading Session List**
  Refer to BTS2 FIX Specification – Market Data, Section 9.5
- **Trading Session List Update Report**
  Refer to BTS2 FIX Specification – Market Data, Section 9.6
- **Security List Request**
  Refer to BTS2 FIX Specification – Market Data, Section 9.7
- **Security List**
  Refer to BTS2 FIX Specification – Market Data, Section 9.8
- **Security List Update**
  Refer to BTS2 FIX Specification – Market Data, Section 9.9

Appendix A - Standard Header and Trailer

A.1 Standard Header

The standard message header format is as follows.

Table 29 – Standard Message Header

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

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

A.2 Standard Trailer

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

The standard message trailer format is as follows.

Table 30 – Standard Message Trailer

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

Appendix B - Component Blocks

B.1 Instrument (symbology) Component Block

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

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

- SecurityID (48)
- SecurityIDSource (22)
- SecuritySubType (762)
- SecurityGroup (1151) for Market Data Request (V) only

The SecurityStatus, SecurityList and SecurityDefinition responses will return the following tags: 22, 48, 55, 106, 107, 223, 224, 225, 762 and 1227. SecurityList and SecurityDefinition will additionally return the following: 201, 202, 454, 455, 456, 470, 541, 350, 351 and 1151.

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

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

Table 31 – Instrument Component Block

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
55	Symbol	N	Ticker symbol or human readable representation of the security. In BTS2, this is the SecShortName.	String
48	SecurityID	N	Unique marketplace assigned identifier number for an order book. Required for inbound transactions to the Exchange except for OrderCancelReplaceRequest(G), OrderCancelRequest(F) and Order Status Request(H).	String
22	SecurityIDSource	N	Identifies class or source of the SecurityID (48) value. Required if SecurityID is specified.	String
454	NoSecurityAltID	N	Number of lternate Security Identifies. Always 1 if presented	NumInGroup
455	SecuirtyAltID	N	ISIN Code	String
456	SecurityAltIDSource	N	4 – ISIN number	String
1151	SecurityGroup	N	Indicates the market classification. For Bursa Malaysia, these values (markets) are valid: MAIN – Main Market ACE – Ace Market ETF – Exchange Traded Fund STRW – Structured Warrants BOND – Bonds/Loans LEAP – LEAP Market	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
762	SecuritySubType	N	In BTS2, this field is used to specify board on which SecurityID is listed. This fiels is equivelant to MarketSegmentID (1300) in Security List (y) and Security Definition (d). Valid values are : NM – NORMAL OD – ODDLOT BI – BUYIN DB – DBT IN – INDEX	String
106	Issuer	N	Issuer of security	String
107	SecurityDesc	N	Optional textual description of the security.	String
223	CouponRate	N	For Fixed Income.	Percentage
224	CouponPaymentDate	N	Date interest is to be paid. Used in identifying Corporate Bond issues.	LocalMktDate
225	IssueDate	N	The date when a bond or stock offering is issued.	LocalMktDate
470	CountryOfIssue	N	ISO Country code of instrument issue (e.g. the country portion typically used in ISIN). Can be used in conjunction with non-ISIN SecurityID (e.g. CUSIP for Municipal Bonds without ISIN) to provide uniqueness.	Country
541	MaturityDate	N	Specifies the maturity date or expiry date of a option.	LocalMktDate
202	StrikePrice	N	Strike Price for an Option.	Price
201	PutOrCall	N	Indicates whether an option contract is a put or call	Int
350	EncodedSecurityDescLen	N	Must be set if EncodedSecurityDesc field is specified and must immediately precede it.	Length
351	EncodedSecurityDesc	N	Encoded (non-ASCII characters) representation of the SecurityDesc field in the encoded format specified via the MessageEncoding field.	Data
1227	ProductComplex	N	Identifier for sector.	String

B.2 InstrumentExtension Component Block

The InstumentExtension Block contains AttrbGrp repeating group to provide addintional instrument atttibutes. Please refer Appendix C for valid values in InstrAttribType (871).

Table 32 – InstrumentExtension Component Block

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
Start of Component block, expanded in line < AttrbGrp >					
870	NoInstrAttrib		Y	Number of repeating InstrAttrib group entries.	NumInGrp
→	871	InstrAttribType	Y	Code to represent the type of instrument attribute	Int
→	872	InstrAttribValue	N	Attribute value appropriate to the InstrAttribType (871) field.	String
End of Component block, expanded in line < AttrbGrp >					

B.3 UnderlyingInstrument Component Block

The UnderlyingInstrument component block, like the Instrument component block, contains all the fields commonly used to describe a security or instrument.

In the case of the UnderlyingInstrument component block it describes an instrument which underlies the primary instrument. Refer to the Instrument component block comments as this component block mirrors Instrument, except for the noted fields.

In the case of an Index, the UnderlyingInstrument component block contains the constituent securities of the Index.

Table 33 - UnderlyingInstrument Component Block

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
309	UnderlyingSecurityID	N	Underlying security's security ID.	String
305	UnderlyingSecurityIDSource	N	Underlying security's SecurityIDSource.	String

B.4 Parties Component Block

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

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

Table 34 –Parties Component Block

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

PARTYROLE(452)	SIZE	COMMENTS	INBOUND MESSAGE ALLOWED
EnteringTrader (36)	30		
EnteringFirm (7)	30		
ContraTrader (37)	30		
ContraFirm (17)	30		

ExecutingTrader (12)	30		TradeCaptureReport (for counterparty only)
ExecutingFirm (1)	30		
OrderEntryOperatorID (44)	30		
OrderOriginationTrader (11)	20	Dealer ID (OMS Id that identifies trader)	NewOrderSingle NewOrderCross OrderCancelReplaceRequest TradeCaptureReport
ClientID (3)	24	Free text	NewOrderSingle NewOrderCross OrderCancelReplaceRequest TradeCaptureReport (own side only)
ClearingFirm (4)	30		

B.4.1 Examples

Firm and individual User for whom the transaction applies:

- Broker Firm (for outbound messages)

  PartyID = "…" – the identifier of the firm

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "1" – Executing Firm

- User (for outbound messages)

  PartyID = "…" – the identifier of the user

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "12" – Executing Trader

- DealerID (for inbound and outbound messages)

  PartyID = "…" – the dealer ID

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "11" – Dealer ID

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

- Broker Firm (For outbound messages)

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

  PartyIDSource = "…" – the type of identifier used

  PartyRole = "7" – Entering Firm

- User (for outbound messages)

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

PartyIDSource = "…" – the type of identifier used

PartyRole = "36" – Entering Trader

B.5 YieldData Component Block

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

Table 35 – YieldData Component Block

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
235	YieldType	N	Type of yield.	String
236	Yield	N	Yield percentage	Percentage

B.6 TriggeringInstruction Component Block

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

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

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

Table 36 – TriggeringInstruction Component Block

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

B.7 LinesofTextGroup Component Block

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

Table 37 – LinesofTextGroup Component Block

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

B.8 SideCrossOrdModGrp Component Block

The SideCrossOrdModGrp component block is used to submit orders on both sides of a crossing order.

Table 38 - SideCrossOrdModGrp Component Block

TAG	FIELDNAME		REQ'D	COMMENTS	FORMAT
552	NoSides		Y	Must be 2	NumInGrp
→	54	Side	Y	Side of order	Char
→	11	ClOrdID	Y	Unique identifier of the order as assigned by institution or by the intermediary with closest association with the investor.	Length
→	Component block <Parties>		N	Insert here the set of "Parties" (firm identification) fields	
→	1	Account	Y	Specifies Investor account.	String
→	38	OrderQty	Y	Quantity ordered. This value represents the number of shares for equities or par, face or nominal value for Fixed Income instruments.	Qty
→	528	OrderCapacity	N	Designates the capacity of the firm placing the order	Char
→	529	OrderRestrictions	Y	For order tagging purpose. See Appendix B. Maximum length is 5 characters.	MultipleCharValue
→	58	Text	N	Free format text string	String

Appendix C - Field Enumerations Sorted By Tag Name

Table 39 – Field Enumerations Sorted By Tag Name

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
266	AggregatedBook	N	Specifies whether or not book entries should be aggregated. Valid values: Y – book entries to be aggregated N – book entries should not be aggregated	Boolean
1057	AggressorIndicator	N	Used to identify whether the order initiator is an aggressor or not in the trade. Valid values: Y – Order initiator is aggressor N – Order initiator is passive	Boolean
380	BusinessRejectReason	Y	Valid values: 0 – Other 1 – Unknown ID 2 – Unknown Security 3 – Unknown Message Type 4 – Application not available 5 – Conditionally required field missing 6 – Not Authorized	Int
292	CorporateAction	N	Identifies the type of Corporate Action that triggered the update. Also referred to as 'Basis of Quotation'. Valid values: A – Ex-Dividend B – Ex-Distribution C – Ex-Rights D – New E – Ex-Interest F – Cash Dividend G – Stock Dividend (Cum Divided) H – Stock Split I – Reverse Stock Split L – Liquidation Reorganization M – Merger Reorganization N – Rights Offering (Cum Rights) P – Spinoff R – Warrant S – Special Action *** BTS2 Specific *** a – Cum Bonus b – Cum Demerge c – Cum Interest d – Cum Listing e – Cum Right of Conversion f – Call Paid g – Cum Delisting h – Offer Closing	MultipleCharValue

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			i – Unlisted j – Ex Bonus k – Ex Demerge l – Ex Listing m – Ex Merge n – Ex Right of Conversion o – Ex Split p – Ex Delisting	
102	CxlRejReason	N	Identifies the reason for the cancel rejection. Valid values: 1 – Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) 99 – Other. Refer to returned Text (58) field for exact reason for rejection.	Int
434	CxlRejResponseTo	Y	Identifies the type of request that a Cancel Reject is in response to. Valid values are: 1 – Order Cancel Request 2 – Order Cancel/Replace Request	Char
1138	DisplayQty	N	Replaces 'MaxFloor' and specifies the disclosed volume on hidden/iceberg orders. This is a V5.0 tag value.	Qty
127	DKReason	Y	Reason for execution rejection. Valid values: A – Unknown Symbol B – Wrong Side C – Quantity Exceeds Order D – No Matching Order E – Price Exceeds Limit F – Calculation Difference Z – Other	Char
18	ExecInst	N	Instructions for order handling. Valid values: G – All or None. o – Withdraw on log off.	Char
150	ExecType	Y	Type of Execution being reported. Describes the specific ExecutionRpt (i.e. Pending Cancel) while OrdStatus (39) will always identify the current order status (i.e. Partially Filled) Valid values: 0 – New 3 – Done for day 4 – Cancelled 5 – Replaced 6 – Pending Cancel 7 – Stopped 8 – Rejected 9 – Suspended C – Expired	Char

TAG	FIELDNAME	REQ'D	COMMENTS		FORMAT
			F – Trade (partial fill or fill) H – Trade Cancel G – Trade Correct I – Order Status U – Order is Unplaced		
432	ExpireDate	Y/N	Conditionally required if TimeInForce = GTD and ExpireTime is not specified.		LocalMktDate
126	ExpireTime	Y/N	Conditionally required if TimeInForce = 'Good till Date/Time'. Currently not in use.		UTCTimeStamp
871	InstrAttribType		Specifies the type of instrument attribute. Valid values are: *** BMB Specific ***		Int
			VALUE	DESCRIPTION	TAG 872
			101 102 103 104 105 106 107 109 110 111 112 113 114 115 118 120	Security Category Par Value Currency code for par value of instrument 30 character AFC name for issuing company Currency code for issue price of bond or warrant Issue price for an instrument (bond, new issue, right, or warrant) Code (ISO3A norm) for the country of listing Short Sell Indicator Industry (Not Used) Delivery Basis 0=Buying-In(T+0) 2=Designated Basis(T+2) 3=Ready Basis(T+3) 4=Immediate Basis(T+2) Shariah Compliant Practice note Date of first day of trading instrument Issued Quantity Maxiumum RSS traded percentage authorized for the trading day Foreign Limitiation Indicator	String Price String String String Price String "2", "P", "R" String Int "Y" String LocalMktDate Qty Int "Y"
1324	ListUpdateAction	N	Specifies the action for a security list. If provided then the Instrument occurrence has explicitly changed: Valid values are: A – Add D – Delete M – Modify		Char
264	MarketDepth	Y	Depth of market for Book Snapshot / Incremental updates.		Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Valid values: 0 – full book depth 1 – top of book 2 and above – book depth (number of levels) Note: For market by price (MBP) this is limited to a depth of five. For market by order (MBO) or non-aggregated book this is limited to ten (10).	
1395	MarketUpdateAction	N	Specifies the action taken for the specified MarketID(1301) + MarketSegmentID(1300). Valid values are: A – Add D – Delete M – Modify	Char
1369	MassActionReportID	Y	Unique Identifier for the Order Mass Action Report. This is a V5.0 tag value.	String
1373	MassActionType	Y	Specifies the type of mass action requested. Valid values: 3 – Cancel orders *** BTS2 Specific *** 100 – Order Status	Int
1374	MassActionScope	Y	Specifies scope of Order Mass Action Request. Valid values: 1 – All orders for a security 7 – All orders 9 – All orders for a Market Segment	Int
1375	MassActionResponse	Y	Indicates the action taken by the counterparty order handling system as a result of the Action Request 0 – Request rejected. 1 – Accepted	Int
1376	MassActionRejectReason	N	Indicates why Order Mass Action Request was rejected. Required if MassActionResponse = 0. Valid values: 0 – Mass Action Not Supported 1 – Invalid or unknown security 8 – Invalid or unknown Market Segment 7 – Invalid or unknown Market 99 – Other	Int
574	MatchType	N	The point in the matching process at which this trade was matched. Valid values: 1 – One-Party Trade Report (privately negotiated trade) 2 – Two-Party Trade Report (privately	String

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			negotiated trade) 4 – Auto-match	
269	MDEntryType	Y	Must be first field in repeating group. This is a list of all the types of Market Data Entries that the firm requesting the Market Data is interested in receiving. For market data requests the following are valid values: 0 – Order information – requests all order related information in market by order and market by price messages (e.g. bids, offers, price, quantity, depth, etc.). 2 – Trade information – returns all trade information and statistics. 3 – Index information – returns all index related information. '*' – Security Statistics (BTS2 specific) – returns security specific market statistics. For responses to market data requests the following are valid values: Valid values: 0 – Bid 1 – Offer 2 – Trade 3 – Index Value 4 – Opening Price 5 – Closing Price 6 – Settlement Price 7 – Trading Session High Price 8 – Trading Session Low Price 9 – Trading Session VWAP Price B – Trade Volume C – Open Interest E – Simulated Sell Price F – Simulated Buy Price P – Early Prices J – Empty Book j - No Trades Exist For 2, 3, 4, 5, 7, 8 the MarketDepth (264) must be set to '1' = 'Top of Book'. *** BTS2 Extensions *** u – Unadjusted previous closing price s – Day's Short Sell t – Days' Proprietary Day Trading (Short Sell Exempt)	Char
281	MDReqRejReason	N	Reason for the rejection of a Market Data request. Valid values: 0 – Unknown symbol	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			1 – Duplicate MDReqID 2 – Insufficient Bandwidth 3 – Insufficient Permissions 4 – Unsupported Subscription Request Type 5 – Unsupported MarketDepth 6 – Unsupported MDUpdateType 8 – Unsupported MDEntryType 9 – Unsupported TradingSessionID	
279	MDUpdateAction	Y	Must be first field in this repeating group. Valid values: 0 – New 1 – Change 2 – Delete	Char
265	MDUpdateType	N	Required if SubscriptionRequestType = Snapshot + Updates (1). Specifies the type of Market Data update. Valid values: 1 – Incremental refresh	Int
528	OrderCapacity	N	Designates the capacity of the firm placing the order. Valid values are: A – Agency M – Market Maker P – Principal R – Riskless Principal	Char
529	OrderRestrictions	Y	Restrictions associated with an order. If more than one restriction is applicable to an order, this field can contain multiple instructions separated by space. Valid values are: 9 – ASEAN Link E – Algorithmic *** BMB Extensions *** I – Internet M – DMA R – Broker Assisted	MultipleCharValue
103	OrdRejReason	N	For optional use with ExecType = 8 (Rejected). Code to identify reason for order rejection. Valid values are: 5 – Unknown order 6 – Duplicate order (e.g. duplicate CLOrdID) 99 – Other. Refer to returned Text (58) field for exact reason for rejection.	Int
378	ExecRestatementReason		Code to identify reason for order cancel or expire.Valid values are: 0 - GT Corporate actions 3 - Expired Carried forward GT orders	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			outside threshold 6 - Order Expired due to Dynamic Limit 99 - Others	
39	OrdStatus	Y	Describes the current state of an order. Valid values are: 0 – New 1 – Partially filled 2 – Filled 4 – Cancelled 5 – Replaced 8 – Rejected 9 - Suspended C – Expired *** BTS2 Defined *** ~~U – Order is Unplaced~~ X – Order with trigger in the book but not active (e.g. Order has not been triggered). ~~Z – Private Order~~	Char
40	OrdType	Y	Indicates the type of order. Valid values are: 1 – Market – The Price (44) field is not used, the order executes against the best prices order on the opposite side. 2 – Limit – The Price (44) field is specified and the order will execute at this price or better. 3 – Stop/Stop Loss – A type of market order that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). The Price (44) field is not specified, but the TriggerPrice (1102) is. The order will be activated as a Market order when the TriggerPrice is reached. 4 – Stop Limit – A type of limit order that is entered into the book when the defined stop price is reached (i.e. a last trade is at or better than that price). Specifies both the Price (44) and the TriggerPrice (1102) field. The order will be activated as a Limit order (using the specified Price as the limit price) when the TriggerPrice is reached. *** BMB Defined *** Z – Market at best	Char
447	PartyIDSource	N	Used to identify class source of PartyID value. Required if PartyID is specified.	Char

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			Required if NoPartyIDs > 0. Valid values are: C – Participant identifier	
452	PartyRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0. Valid values are: 1 – Executing Firm 3 – Client ID 4 – Clearing Firm 7 – Entering Firm 11 – Order origination trader 12 – Executing Trader 13 – Order origination firm 17 – Contra Firm 36 – Entering trader 37 – Contra trader 44 – Order Entry Operator ID	Int
22	SecurityIDSource	N	Identifies class or source of the SecurityID (48) value. Valid values: 99 – Marketplace assigned identifier	String
559	SecurityListRequestType	N	Specifies the criteria of the request: 0 – Symbol 4 – All Securities 5 – MarketID (Specific Market)	Int
560	SecurityRequestResult	N	Result of the Security Request identified by the SecurityReqID. Valid values: 0 – Valid request 1 – Invalid or unsupported request 2 – No instruments found that match selection criteria 3 – Not authorized to retrieve instrument data 4 – Instrument data temporarily unavailable 5 – Request for instrument data not supported *** 100+ BTS2 Specific*** 100 – Invalid MarketID	Int
321	SecurityRequestType	N	Type of Security Definition Request. Valid values: 4 – Symbol (security ID in BTS2) 8 – All Securities 9 – MarketID	Int
323	SecurityResponseType	N	Type of Security Definition response. Valid values: 4 – List of securities returned per request 5 - Reject security proposal 6 – Cannot match selection criteria	Int
326	SecurityTradingStatus	N	Identifies the trading status applicable	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			to the transaction. Valid values: 2 – Trading suspended 17 – Ready to trade 18 – Not available for trading 20 – Unknown or Invalid	
980	SecurityUpdateAction	N	Specifies the update action for the security. Valid values: A – Add D – Delete M – Modify	Char
54	Side	Y	Optional qualifier used to indicate the side of the market. Valid values are: 1 – Buy 2 – Sell 5 – Regulated Short Sell(RSS) 6 – Proprietary Day Trading(PDT) I - Intraday Short Sell(IDSS) V – Permitted Short Sell(PSS)	Char
263	SubscriptionRequestType	N	Used to subscribe for Quote Status Report messages. Subscribe or unsubscribe for security status to security specified in request. Subscription type request. Valid values are: 0 – Snapshot 1 – Snapshot+Updates (Subscribe) 2 – Disable previous Snapshot+Update Request (unsubscribe)	Char
274	TickDirection	N	Direction of the "tick". Valid values: 0 – Plus Tick 1 – Zero-Plus Tick 2 – Minus Tick 3 – Zero-Minus Tick	Char
59	TimeInForce	N	Indicates time in force techniques that are valid for the specified market segment. Valid values are: 0 – Day 1 – Good till cancelled 2 – At the Opening (Session) 3 – Immediate or Cancel (IOC) 4 – Fill or Kill (FoK) 6 – Good till date 7 – At the Close(Session) S – Session (This field used as **outbound** (to the marketplace) when use 59=2 or 59=7. **This field not allow for inbond**.)	Char
751	TradeReportRejectReason	N	Reason for Rejection of Trade Report Valid Values: 99 – other	int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
487	TradeReportTransType	N	Identifies Trade Report message transaction type 0 – New 1 – Cancel 2 – Replace	Int
856	TradeReportType	N	Type of Trade Report 0 – Submit 1 – Alleged 2 – Accept 3 – Decline 6 – Trade Report Cancel 10 – Pended	Int
749	TradeRequestResult	Y	Result of Trade Request. Valid Values: 0 – Successful 8 – TradeRequestType not supported 99 – Other	Int
750	TradeRequestStatus	Y	Status of Trade Request. Valid Values: 0 – Accepted 1 – Completed 2 – Rejected	Int
569	TradeRequestType	Y	Type of Trade Capture Report. Valid values: 0 – All Trades	Int
336	TradingSessionID	Y	Identifier for Trading Session. A trading session spans an extended period of time that can also be expressed informally in terms of the trading day. Valid values are: EN – Enquiry POP1 – Pre-Opening OPN1 – Opening CNT1 – Trading BRK1 – Break POP2 – Second Pre-Opening CNT2 – Second Trading POC2 – Pre-Closing CLS2 – Closing TAL2 – Trading at last EOT – Close EOD – End Of Day HLT – Halt CBH – Circuit Breaker Halt	String
340	TradSesStatus	Y	State of trading session. Valid values are: 6 – Request rejected *** BTS2 Specific *** 100 – Pending – Indicates that trading session has not been started 101 – Triggered – Indicates trading session has either occurred or is	Int

TAG	FIELDNAME	REQ'D	COMMENTS	FORMAT
			the current session 102 – Deleted – This trading session has been removed from the trading schedule.	
567	TradSesStatusRejReason	N	Used with TradSesStatus = "Request Rejected". Valid values are: 1 – Unknown trading session id	Int
1327	TradSesUpdateAction	N	Specifies the action taken for the specified trading sessions. Valid values: A – Add D – Delete M – Modify	Char
939	TrdRptStatus	N	Status of Trade Report 0 – Accepted 1 – Rejected	int
828	TrdType	N	Type of Trade. 0 – Regular Trade 22 – Privately Negotiated Trades *** BMB Extensions *** 100 – Crossing Order Trade	Int
61	Urgency	N	Urgency Flag. Valid values are: 0 – Normal 1 – Flash 2 – Background	Char
236	Yield	Y/N	Yield percentage – This is the yield equivalent of the price. Price (44) and Yield (236) are mutually exclusive. Currently not in use.	Percentage
235	YieldType	N	Supported values are: OPENAVG – Open Average Yield CLOSE – Closing Yield High – Trading session high yield Low – Trading session low yield Last – Last yield WAvg – Weighted Average Change – Change from reference yield.	String

Appendix D - FIX Data Types

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

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

char	Single character value, can include any alphanumeric character or punctuation except the delimiter. All char fields are case sensitive (i.e. m != M). The following fields are based on char.	
	Boolean	char field containing one of two values: 'Y' = True/Yes 'N' = False/No
String	Alpha-numeric free format strings, can include any character or punctuation except the delimiter. All String fields are case sensitive (i.e. morstatt != Morstatt).	
	MultipleCharValue	string field containing one or more space delimited single character values (e.g. |18=2 A F|).
	MultipleStringValue	string field containing one or more space delimited multiple character values (e.g. |277=AV AN A|).
	Country	string field representing a country using ISO 3166 Country code (2 character) values (see FIX Specifications Volume 6 - Appendix 6-B).
	Currency	string field representing a currency type using ISO 4217 Currency code (3 character) values (see FIX Specifications Volume 6 - Appendix 6-A).
	Exchange	string field representing a market or exchange using ISO 10383 Market Identifier Code (MIC) values (see FIX Specifications Volume 6 - Appendix 6-C).
	MonthYear	string field representing month of a year. An optional day of the month can be appended or an optional week code. Valid formats: YYYYMM YYYYMMDD YYYYMMWW Valid values: YYYY = 0000-9999; MM = 01-12; DD = 01-31; WW = w1, w2, w3, w4, w5.
	UTCTimestamp	string field representing Time/date combination represented in UTC (Universal Time Coordinated, also known as "GMT") in either YYYYMMDD-HH:MM:SS (whole seconds) or YYYYMMDD-HH:MM:SS.sss (milliseconds) format, colons, dash, and period required. Valid values: * YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second) (without milliseconds). * YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds). Leap Seconds: Note that UTC includes corrections for leap seconds, which are inserted to account for slowing of the rotation of the earth. Leap second insertion is declared by the International Earth Rotation Service (IERS) and has, since 1972, only occurred on the night of Dec. 31 or Jun 30. The IERS considers March 31 and September 30 as secondary dates

	for leap second insertion, but has never utilized these dates. During a leap second insertion, a UTCTimestamp field may read "19981231-23:59:59", "19981231-23:59:60", "19990101-00:00:00". (see http://tycho.usno.navy.mil/leapsec.html)
UTCTimeOnly	string field representing Time-only represented in UTC (Universal Time Coordinated, also known as "GMT") in either HH:MM:SS (whole seconds) or HH:MM:SS.sss (milliseconds) format, colons, and period required. This special-purpose field is paired with UTCDateOnly to form a proper UTCTimestamp for bandwidth-sensitive messages. Valid values: HH = 00-23, MM = 00-60 (60 only if UTC leap second), SS = 00-59. (without milliseconds) HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds).
UTCDateOnly	string field representing Date represented in UTC (Universal Time Coordinated, also known as "GMT") in YYYYMMDD format. This special-purpose field is paired with UTCTimeOnly to form a proper UTCTimestamp for bandwidth-sensitive messages. Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31.
LocalMktDate	string field representing a Date of Local Market (as opposed to UTC) in YYYYMMDD format. This is the "normal" date field used by the FIX Protocol. Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31.
data	string field containing raw data with no format or content restrictions. Data fields are always immediately preceded by a length field. The length field should specify the number of bytes of the value of the data field (up to but not including the terminating SOH). **Caution**: The value of one of these fields may contain the delimiter (SOH) character. Note that the value specified for this field should be followed by the delimiter (SOH) character as all fields are terminated with an "SOH".