Johannesburg Stock Exchange

Trading and Information Solution

JSE Specification Document

Volume 04 – Drop Copy Gateway (FIX 5.0 SP2)

Version	3.05
Release Date	01 August 2019
Number of Pages	48 (Including Cover Page)

1 DOCUMENT CONTROL

1.1 Table of Contents

1.2 Document Information

Drafted By	JSE
Status	Final
Version	3.05
Release Date	01 August 2019

1.3 Revision History

Date	Version	Description
08 July 2011	1.00	Initial Draft
30 November 2011	1.01	Final
05 July 2013	2.00	Functionality Updates related to the 2013 product upgrade
22 August 2014	2.01	Introducing the ability to submit Exclude Hidden Limit Orders
29 February 2016	3.00	Integrated Trading and Clearing Project changes. Equity Market Enhancements: • Hidden Order functionality enhanced • Introduction of On Book Cross Order Trade • Introduction of EOD Volume Auction
4 August 2016	3.01	Description of Working Indicator field updated
6 March 2017	3.02	Description of Multi Leg Reporting Type (442) field updated.
19 April 2018	3.03	3.5.2.5. Details of SecondaryTradeReportID (818) dissemination added. 7.5.1 New field SecondaryTradeReportID (818) added to Execution Report message
2319 July 2019	3.04	3.5.3 Details regarding the assigning of the ClOrdID (11) of the base order to Implied Out and Implied match orders added. 7.5.1 Possible values for Cross Type (549) field added
01 August 2019	3.05	1.5 Definition for EHL removed
		3.5.2.3 Details of Public Order IDs added
		7.5.1 New field MDEntryID (278) added to the Execution Report message

1.4 References

FIXT 1.1 Specification
FIX 5.0 (Service Pack 2) Specification
Volume 01 – Native Trading Gateway Specification
Volume 02 – FIX Trading Gateway (FIX 5.0 SP2) Specification

1.5 Contact Details

JSE Limited Trading and Market Services Division One Exchange Square Gwen Lane, Sandown South Africa Tel: +27 11 520 7000 www.jse.co.za	**Trading and Market Services ITAC Queries** Email: CustomerSupport@jse.co.za

1.6 Definitions, Acronyms and Abbreviations

Client	A member firm connected to the drop copy gateway.
FIX	Version 5.0 (Service Pack 2) of the Financial Information Exchange Protocol.
FIX Connection	A bi-directional stream of ordered messages between the client and server within a particular login. A FIX connection ends when the client logs out or if the TCP/IP connection is terminated.
FIX Session	A bi-directional stream of ordered messages between the client and server within a continuous sequence number series. A single FIX session can exist across multiple FIX connections.
FIXT	Version 1.1 of the Financial Information Exchange Session Protocol.
JSE	Johannesburg Stock Exchange
Trading Gateway	The interface at the JSE for the equities and derivatives/bonds Markets that allows member firms to submit and manage orders and trades.
Server	The drop copy gateway at the JSE for the equities and derivatives/bonds markets.
Trader	Each order request must be submitted under a particular trader.
Trader Group	Trader group within a Firm which the Trader belongs to.
Repeating Group	A set of related attributes which occur more than once within a field of a message
NSX	The Namibian Stock Exchange
Open Order	Identifies an order which has a remaining quantity in the order book. An amendment or a cancellation can be done for an Open Order.
Parked Order	Identifies an order which is not yet been activated. GFA, GFX, ATC orders will be parked until the relevant auction call phase is started. Unelected Stop and Stop Limit orders will be parked until the stop price is reached.
Client Account	This is the Client Account as reflected in the JSE back office system for whom the order is submitted by the Firm.
Visible Order	Identifies an order that is visible to the market. The order has a Disclosed Quantity that is equal to Order Quantity.
Passive Order	An order residing in the order book.
MES	Minimum Execution Size is the minimum volume of the Hidden Pegged and Pegged Limit order which is permitted to execute.
MRS	Minimum Reserve Size (is the equivalent of the JSE's MOS – for Hidden Pegged and Pegged Limit orders) MRS is the minimum order volume for orders to qualify as Hidden Pegged and Pegged Limit orders.
~~**EHL**~~	~~Exclude Hidden Pegged and Pegged Limit order is an order that will only match with existing Visible Orders in the order book and will expire if the remainder is not filled within a period of time defined. At no point will these orders execute with Hidden Pegged and Pegged Limit orders.~~

2 OVERVIEW

The JSE offers a drop copy gateway for the equities and derivatives/bonds markets that will enable members firms to receive additional copies of the Execution Reports generated by the system. This interface may also be used by clients to download the current status of all their active orders in the event of a failure. The drop copy service cannot be used to submit, amend or delete orders or receive market data.

The interface is a point-to-point service based on the technology and industry standards TCP/IP, FIXT and FIX. The session and application event models and messages are based on versions 1.1 and 5.0 (Service Pack 2) of the FIXT and FIX protocols respectively.

The encryption of messages between the client and server is not supported.

3 SERVICE DESCRIPTION

3.1 Services Supported by Trading Gateway

A description of the services (Example: order types, notification of market operations actions) available via the Trading Gateways is provided in the Trading Gateway (FIX 5.0 SP2) and Native Trading Gateway documents which clients are encouraged to read together with this specification.

3.2 Connection Configuration

3.2.1 Real-Time Connections

A client that is enabled for this drop copy service will receive a drop copy of each eligible Execution Report immediately after it is published.

A member firm connection will be configured to receive a drop copy of all the Execution Report messages generated for the firm for the events outlined in Section 3.3. If required, a member firm connection could be configured to only receive real time updates for selected interface users and/or for a selected set of instruments.

For the purpose of redundancy, the service supports the configuration of multiple drop copy connections to send the same information on the activity of the firm or trader and trader group combinations.

The identity of the Interface User ID (CompID) that transmitted the order a particular drop copy relates to will be specified in the header field OnBehalfOfCompID (115).

Please refer to Sections 6.4 and 0 for a description of how the Execution Reports published during the time a client is disconnected from the server, may be recovered.

A client may also use the own order download service (outlined in Section 3.3.1) to recover the status of all active orders in the event of a client system failure.

3.2.2 Non-Real Time Connections

Execution Reports will not be streamed real time for clients enabled on this service. Such a client may only connect to the server to use the own order download service outlined in Section 3.3.1.

3.3 Supported Events

Clients will receive drop copies of the Execution Reports generated for the following events that occur in all the markets:

(i) Order accepted
(ii) Order rejected
(iii) Order executed
(iv) Quote executed
(v) Order expired
(vi) Order cancelled
(vii) Order cancel/replaced
(viii) Trade cancellation
(ix) Trade correction

3.3.1 Quotes

As Quotes functionality is enabled only for the Derivative and Bond Markets, this section is not applicable for the members of the JSE and NSX Equity Markets.

The Quote Status Report, Quote Request Reject and Mass Quote Acknowledgement messages that are sent by the Trading Gateway to acknowledge or reject Quotes, Mass Quotes and Quote Cancel messages are not available via the drop copy service.

However, the Execution Reports sent when quotes are executed are available as drop copies. The ClOrdID (11) of such a message will contain the QuoteMsgID (1166) of the last Quote message or QuoteID (117) of the last Mass Quote message that updated the executed quote. The side, quantity and price fields (e.g. Side (54), LastQty (32), LastPx (31), LeavesQty (151), OrderQty (38), Price (44), etc.) will contain information for the executed side. As the matching system does not keep track of cumulative quantity for quotes, the value in the field CumQty (14) will be '0'.The CumQty (14) will be zero in the Execution Reports that correspond to leg instrument trades when a quote for a multi-legged instrument executes.

3.3.2 Private Request for Quotes

As privately negotiated RFQs functionality is enabled only for the derivatives and bonds markets, this section is not applicable for the members of the equity market.

In the process of privately negotiated RFQs, the requester requests for quotes using the Quote Request message. The Quote Request will then be directed to an intended set of market makers. The market maker can respond to the RFQ via submitting a Quote or respond with a rejection via a Quote Request Reject message. The requester can then accept a Quote submitted by one of the market makers, which matches his interests.

During the private RFQ negotiation process the Execution Reports generated due to the following events will be available via the drop copy gateway.

(i) Quote Acceptance

(ii) Quote Rejection

(iii) Quote Cancellation

(iv) Quote Expiration

(v) Quote Execution

The Execution Reports generated due to executions will be available via drop copy users of the requester's firm as well as the market maker's firm. The Execution Reports generated due to the rest of the events above will only be available via the drop copy users of the market maker's firm.

3.4 Open Order Download

Any client may use the Mass Order Status Request message to download the current status of each active order and quote side for a specified firm or trader and trader group combination. The total number of Mass Order Status Requests that a client may submit is limited to MAX_OOBD_REQUESTS_PER_USER *<1000>* each day. A client may request the JSE to reset its request count. This feature is intended to help manage an emergency situation and should not be relied upon as a normal practice.

The Mass Order Status Request may be submitted by Clients to receive all active orders for the following:

- Active orders for the firm OR trader and trader group combination only

  Mass Order Status Request with the MassStatusReqType (585) of "8" and the relevant PartyID combination specified in Section 3.5.5

- Active orders for a specified instrument and Party ID combination

  Mass Order Status Request with the MassStatusReqType (585) of "1" and the relevant PartyID combination specified in Section 3.5.5

- Active orders for a specified segment and Party ID combination

  Mass Order Status Request with the MassStatusReqType (585) of "100" and the relevant PartyID combination specified in Section 3.5.5

If a request is successful, the server will respond with an Execution Report for each active order for the specified trader and trader group combination. Each such message will include the MassStatus ReqID (584) of the request, an ExecID (17) of "0" and an ExecType (150) of Order Status (I). The last Execution Report sent by each partition in response to the request will include a LastRptRequested (912) of Last Message (Y).

The server will transmit a single Execution Report if the request is rejected or if there are no active orders for the specified trader and trader group combination. Such a message will include the MassStatusReqID (584) of the request, an ExecID (17) of "0", an ExecType (150) of Order Status (I) and an OrdStatus (39) of Rejected (8). The message will not include fields that relate to order-specific information (e.g. ClOrdID (11), OrderID (37), OrderQty (38), LeavesQty (151), CumQty (14), OrdType (40), Security ID (48), and SecurityIDSource (22), OrigClOrdID(41), ExecType(150), TradeMatchID(880), OrderStatus(39), WorkingIndicator(636), OrdRejReason(103), Text(58), LastQty(32), LastPx(31), Account(1), TimeInForce(59), ExpireTime(126), ExpireDate(432), Side(54), DisplayQty(1138), DisplayMethod(1084), MinQty(110), Price(44), StopPx(99)). The reason for the rejection will be specified in the field OrdRejReason (103).

A Business Message Reject will be sent to reject a Mass Order Status Request if the server is unable to process it in the unlikely event of a system outage. If the outage occurs before the server has sent all of the messages in response to a Mass Order Status Request, it will terminate the open order download. An Execution Report will be sent if the open order download is terminated. It will include the MassStatusReqID (584) of the request, an ExecID (17) of "0", an ExecType (150) of Order Status (I) and an OrdStatus (39) of Rejected (8).

3.5 Execution Reports

The Execution Report message is used to communicate many different events to clients. The events are differentiated by the value in the ExecType (150) field as outlined below.

Exec Type	Usage	Ord Status
0	**New** Indicates that a new order has been accepted. This message will also be sent unsolicited if an order was submitted by JSE Market Operations on behalf of the client. This message will also be sent when an unelected order is elected and added to the order book without receiving an execution.	0
8	**Rejected** Indicates that an order has been rejected. The reason for the rejection is specified in the field OrdRejReason (103).	8
F	**Trade** Indicates that an order has been partially or fully filled. The execution details (e.g. price and quantity) are specified. This message will also be sent when an unelected order is elected and receives executions on aggression. This message will also be sent when a parked order with time in force GFX/GFA/ATC is un-parked and receives executions on aggression.	1, 2
C	**Expired** Indicates that an order has expired in terms of its time qualifier or due to an execution limit. This message will be sent out when any remaining orders (except GTD and GTC) are expired at market close. Expiration of GTC and GTD orders occur at the market start the next day This message will also be sent when orders are expired based on the cancel on disconnect/log out feature. This message will also be sent when orders are expired upon entering the order book when the number of orders in the order book is at the maximum allowed level. The reason for the expiration is specified in the Text (58) field. This message will also be sent when a Market Order or a Stop Order is expired at the point of aggressing the order book during the Continuous Trading session, if a circuit breaker is breached during that aggression. The reason for the expiration is specified in the Text (58) field.	C
4	**Cancelled** Indicates that an order cancel request has been accepted and successfully processed. This message will also be sent unsolicited if the order was cancelled by JSE Market Operations. In such a scenario the Execution Report will include an ExecRestatementReason (378) of Market Option (8). It will not include an OrigClOrdID (41).	4
5	**Replaced** Indicates that an order cancel/replace request has been accepted and successfully processed.	0, 1
L	**Triggered** Indicates that a parked ATC, GFX, GFA or Stop Order has been activated and moved to the main container. The order is available for execution.	0, 1

9	**Suspended** Indicates that a GFA order that was active has been parked and is no longer available for execution.	0, 1
D	**Restated (Order Cancel/Replace by Market Operations)** Indicates that an order has been amended by JSE Market Operations. The unsolicited message will include an ExecRestatementReason (378) of Market Option (8). It will not include an OrigClOrdID (41).	0, 1
H	**Trade Cancel** Indicates that an execution has been cancelled by JSE Market Operations or by clients. An ExecRefID (19) to identify the execution being cancelled will be included.	0, 1, 4, C
G	**Trade Correct** Indicates that an execution has been corrected. The message will include an ExecRefID (19) to identify the execution being corrected and the updated execution details (e.g. price and quantity).	1, 2, 4, C
I	**Order Status Response** Indicates the current status of an order.	0, 1
8	**Order Status Reject** Indicates that an order mass status request has been rejected.	8
	It should be noted that the Exchange will generally not amend orders or trades. These events are included in the above table for completeness.	

3.5.1 Order Status

As specified in the FIX protocol, the OrdStatus (39) field of an Execution Report is used to convey the current state of an order. If an order simultaneously exists in more than one order state, the value with highest precedence is reported as the OrdStatus (39). The relevant order statuses are given below from the highest to lowest precedence.

Value	Meaning
2	Filled
9	Suspended
4	Cancelled
C	Expired
1	Partially Filled
0	New
8	Rejected

3.5.2 Order and Execution Identifiers

3.5.2.1 Client Order IDs

In the case of orders, the ClOrdID (11) included in each Execution Report will be that specified when the order was submitted. An order's ClOrdID (11) will be updated each time an Order Cancel/Replace Request or an Order Cancel Request is accepted. The transact time field will not be populated on the Execution Report that is sent to the Client as a result of a duplicate ClientOrderID validation.

3.5.2.2 Order IDs

The server uses the OrderID (37) field of the Execution Report to affix the order identification numbers of the trading engine. Order IDs are unique across trading days in perpetuity.

In terms of the FIX protocol, unlike ClOrdID (11) which requires a chaining through Cancel/Replace Requests and Cancel Requests, the OrderID (37) of an order will remain constant throughout its life.

3.5.2.3 Public Order IDs

The server uses MDEntryID (278) field of the Execution Report to affix the Public Order ID of an order. This is an order identification number that will be stamped for each order that has an OrderID (37). For all orders that are not iceberg orders, the MDEntryID (278) will be the same as the OrderID (37). For iceberg orders, the MDEntryID (278) will renew with each replenishment to the visible order size. Participants will be able to identify their orders on the market data feeds using MDEntryID (278) which is the identification number that will be disseminated for order book updates on market data feeds.

~~3.5.2.3~~3.5.2.4 Execution IDs

The server uses the ExecID (17) field to affix a unique identifier for each Execution Report. ExecIDs are unique across trading days in perpetuity. If an Execution Report message is used to notify a client of a trade cancellation or correction, the ExecRefID (19) will refer to the Execution ID generated for the original execution.

~~3.5.2.4~~3.5.2.5 Trade IDs

The server uses the TrdMatchID (880) field to affix a unique identifier for each trade. This identifier is referenced in the Trade Capture Reports published by the post trade system and the trade messages of the FAST and MITCH market data feeds. Trade IDs are unique across trading days in perpetuity.

An Execution Report published to notify a client of a trade cancellation or correction includes the TradeID of the trade.

~~3.5.2.5~~3.5.2.6 Secondary Trade Report IDs

In the case of orders, the SecondaryTradeReportID (818) included in each Execution Report will be the same as specified when the order was initially submitted. An order's SecondaryTradeReportID (818) could be updated each time an Order Cancel/Replace Request is accepted.

3.5.3 Strategies

The values specified in the fields Price (44), StopPx (99) and LastPx (31) for Execution Reports relating to multi-legged instruments may contain negative prices.

If an order for a strategy receives an execution, it will receive an Execution Report for the multi-legged instrument as well as separate Execution Reports for each of the associated leg instruments. The field MultiLegReportingType (442) should be used to determine whether a particular Execution Report relates to the multi-legged instrument or a leg instrument.

While the ClOrdID (11) of an Execution Report for a leg trade will be the same as the ClOrdID (11),QuoteMsgID (1166) or QuoteID (117) of the order or quote for the multi-legged instrument, the OrderID (37) will not be the same. The SecondaryOrderID (198) for a leg trade will contain the OrderID (37) of the associated order or quote for the multi-legged instrument.

Execution Reports corresponding to the leg trades will have a CumQty (14) of zero (0) when a quote for a multi-legged instrument executes.~~,~~

The ClOrdID (11) of the implied orders on the leg instruments will be populated with the same ClOrdID (11) as the explicit order. This will enable clients to track implied orders and implied

order related executions that resulted from their explicit orders. (i.e. The ClOrdID (11) of an Execution Report for a leg implied execution will be the same as that of the order or quote for the multi-legged instrument).

3.5.4 Instrument Identification

Instruments will be identified using the SecurityID (48) field.

It is required to specify SecurityID Source (22) field as well.

The instrument identification included in an Execution Report will be that specified in the order the message relates to.

3.5.5 Party Identification

ID	Description	Relevant FIX Tags
Executing Firm (Firm ID)	Identifier of the trading firm the order is submitted under.	PartyRole (452) = 1 PartyID (448)
Trader Group	Identifier of the Trader Group the order is submitted under.	PartyRole (452) = 76 PartyID (448)
Trader	Identifier of the Trader the order is submitted under. This is the JSE Trader ID.	PartyRole (452) = 53 PartyID (448)
Client Account	Client Account information applicable to an order.	Account (1)
Market Makers	Identifier of the market maker firms to whom a private RFQ is directed at.	PartyRole (452) = 66 PartyID (448)
Contra Firm	The firm on the contra-side of a privately negotiated RFQ.	PartyRole (452) = 17 PartyID (448)
Contra Trader	The trading mnemonic of the contra-side of a privately negotiated RFQ.	PartyRole (452) = 37 PartyID (448)

3.5.6 Quotation Conventions

The values specified in the fields Price (44), StopPx (99), LastPx (31) and AvgPx (6) should be interpreted in terms of the applicable quotation convention for the instrument.

The values specified in these fields should be interpreted as the price per share for equity instruments. They should be interpreted as price per contract for futures and derivative strategies. For a fixed income instruments and options, they should, depending on the applicable convention, be interpreted as percentage of par, discount rate, yield or volatility.

The value, if any, specified in the field LastParPx (669) should always be interpreted as percentage of par.

3.5.7 Fixed Income Instruments

If an order for a fixed income instrument[1]is partially or fully filled, the accrued interest associated with the trade will be included in the field AccruedInterestAmt (159) of the Execution Report. The value in this field will generally be positive indicating the total accrued interest due from the buyer to the seller. However, in the case of trades settled on or after the

[1]Among Fixed Income instruments, only Regular Coupon Bonds and TIPS have accrued interest associated with them

ex-coupon date of an instrument, this value will be negative to indicate the accrued interest due from the seller to the buyer.

Based on the instrument configurations, for fixed income instruments quoted in discount rate or yield, the notification of an execution may include the limit price expressed as a percentage of par in the field ParPx (32032) and traded price expressed as a percentage of par in the field LastParPx (669). The Execution Report may include the implied yield of the order in the ConvertedYield (30005) and trade in the Yield (236) field in the case of instruments quoted in percentage of par.

Fixed Income information will be republished for carried forward orders at the start of the Market. The Execution Report will include an ExecType (150) of Restated (D) and ExecRestatementReason (378) of GT Renewal/Restatement (1) to indicate the automatic re-pricing.

3.6 Timestamps and Dates

The timestamps SendingTime (52), OrigSendingTime (122) and TransactTime (60) are in UTC and in the YYYYMMDD-HH:MM:SS.sss format. ExpireTime (126) is in UTC and in the YYYYMMDD-HH:MM:SS format.

All dates (i.e. MaturityDate (541) and ExpireDate (432)) are in the YYYYMMDD format and specified in the local date for the server (i.e. not in UTC)).

3.7 Repeating Groups (Components/Component Block)

If a repeating group is used in a message, the number of IDs (for example NoPartyIDs field in the trading party repeating group) will be specified first before the repeating group starts. This is applicable for both the messages generated by the client and the server.

The messages generated by the server will have the fields within a repeating group in order as specified in the messages in this document.

The messages generated by a client should have the first field in a repeating group in order as specified in the messages in this document. If the first field in a repeating group is in order, a message generated by a client will be accepted; else the message will be rejected

If the same FIX tag is repeated with different values in the client generated message outside of a repeating group, the server takes the value in the last tag. The server will not reject such messages.

If other client-initiated administrative messages or application messages contain repeated tags outside component blocks, such requests will be rejected by the server. However, if a client-initiated Logon message contains repeated tags; the server may not acknowledge the login request and will not send any reply.

3.8 Generating Reject Messages

If a required tag or a conditionally required tag is missing in a message sent by a client, the server will send a session reject message for that.

Furthermore, if an unsupported value is sent with a tag, an execution report or an order cancel reject is sent by the server.

3.9 Mapping FIX Order ID to MITCH Order ID

To convert FIX Order ID to MITCH Order ID:

Step 1 – Remove leading O (prefix)

Step 2 - Convert using base 62 using the base 62 conversion table below

Step 3 – Convert to binary

Order ID format (binary)

20 bits	2bits	3 bits	2bits	32 bits (4 bytes)
<number of sec>	[0-3]	[0-7]	[0-3]	
The number of 5 mins intervals from Jan 1, 2010)	ID	Partition id	Thread id	Order number

8 bytes

Order ID format (ASCII)

1 byte	11 bytes
O	0-9, A-Z, a-z
Prefix	Base 62 encoded order id

E.g.

OrderID in FIX (ASCII base 62 characters)	O04Xj7Wu76ta
OrderID in MITCH gateway (Binary ID converted to decimal)	61512470073704470

Steps to follow

- Convert using base 62 conversion into decimal as depicted below
- Note: Please refer to the base 62 conversion table attached below

FIX Order ID (ASCIIC haracter)	Correspond ing decimal value	Base 62^x	value	Multiplied decimal value
A	36	62^0	1	36
T	55	62^1	62	3,410
6	6	62^2	3,844	23,064
7	7	62^3	238,328	1,668,296
U	56	62^4	14,776,336	827,474,816
W	32	62^5	916,132,832	29,316,250,624
7	7	62^6	56,800,235,584	397,601,649,088
J	45	62^7	3,521,614,606,208	158,472,657,279,360
X	33	62^8	218,340,105,584,896	7,205,223,484,301,568
4	4	62^9	13,537,086,546,263,552	54,148,346,185,054,208
0	0	62^10	839,299,365,868,340,224	-
				61,512,470,073,704,470

Note

- Please use 64 bit integer data types for the calculation else integers will overflow
- Excel also rounds the value since its using a 64 bit float data type for the calculation

The base 62 mapping table

0	0	20	K	40	e	60	y
1	1	21	L	41	f	61	z
2	2	22	M	42	g		
3	3	23	N	43	h		
4	4	24	O	44	i		
5	5	25	P	45	j		
6	6	26	Q	46	k		
7	7	27	R	47	l		
8	8	28	S	48	m		
9	9	29	T	49	n		
10	A	30	U	50	o		
11	B	31	V	51	p		
12	C	32	W	52	q		
13	D	33	X	53	r		
14	E	34	Y	54	s		
15	F	35	Z	55	t		
16	G	36	a	56	u		
17	H	37	b	57	v		
18	I	38	c	58	w		
19	J	39	d	59	x		

3.10 Mapping Trade Match ID to MITCH Trade ID

To convert FIX Trade Match ID to MITCH Trade ID:

Step 1 – Remove Leading Prefix

Step 2 – Convert using base 62

Step 3 – Convert to binary

Trade ID format (binary)

20 bits	2bits	3 bits	2bits	24 bits
<number of sec>	[0-15]	[0-7]	[0-3]	
The number of 5 mins intervals from Jan 1, 2010)	ID	Partition id	Thread id	Trade number

8 bytes

Trade ID format (ASCII)

10 bytes
0-9, A-Z
Base 62 encoded Trade ID

E.g.:

ASCII trade id for FIX	T5DIF33YV0
Binary trade id (decimal) for MITCH	1138517709214786

Base 62 conversion into decimal as displayed below:

FIX Trade ID (ASCII Character)	Corresponding decimal value	Base 62^x	value	Multiplied decimal value
0	0	62^0	1	0
V	31	62^1	62	1922
Y	34	62^2	3,844	130696
3	3	62^3	238,328	714984
3	3	62^4	14,776,336	44329008
F	15	62^5	916,132,832	13741992480
I	18	62^6	56,800,235,584	1022404240512
D	13	2^7	3,521,614,606,208	45780989880704
5	5	62^8	218,340,105,584,896	1091700527924480
				1138517709214786

4 CONNECTIVITY

4.1 Interface User (CompIDs)

The Interface User (CompID) of each client must be registered with the JSE The JSEfor all the equity and derivatives/bonds Markets before FIX communications can begin through the Gateway. A single client may have multiple connections to the server (i.e. multiple FIX sessions, each with its own Interface User ID (CompID) if it has multiple valid Interface User IDs (CompIDs)).

The Interface User (CompID) of the server will be JSEDCPGW. The messages sent to the server will contain the Interface User (CompID) assigned to the client in the field SenderCompID (49)) and JSEDCPGW in the field TargetCompID (56). The messages sent from the server to the client will contain JSEDCPGW in the field SenderCompID (49) and the Interface User (CompID) assigned to the client in the field TargetCompID (56).

4.1.1 Passwords

The JSE will enable password policies, where each new Interface User ID (CompID) will be assigned a default password on registration. Clients must change the password to one of their choosing via the Logon message with their first Logon message. The acceptance of a login request that includes a password change request indicates that the new password has been accepted. The new password will, if accepted, be effective for subsequent logins. If a new password is rejected, the Text (58) of the Logout message will indicate why the Logon is rejected.

The password policy will also dictate the use of minimum requirements where a password must be at least 8 characters long, include at least 1 special character, 1 alpha character and 1 numeric character. The server will further validate the historic use of passwords and set the number of passwords in history to 60. Clients submitting a Logon message that is the same as any of the previous 60 passwords used will be rejected. The Text (58) of the Logout message will indicate why the Logon is rejected.

4.2 Production IP Addresses and Ports

The IP address of each client must be registered with the System before FIX communications can begin. The IP addresses and ports of the production servers will be available upon request to registered JSE clients.

The JSE will assign each registered client to one of the primary IP addresses and ports and one of the secondary IP addresses and ports.

4.3 Failover and Recovery

The system has been designed with fault tolerance and disaster recovery technology that ensures that trading will continue in the unlikely event of a process or site outage.

If the client is unexpectedly disconnected from the server, it should attempt to re-connect to primary site within a few seconds. The client should only attempt to connect to the secondary IP address and port if so requested by the JSE. Please refer to the separate Client Failover and Recovery Document which will be issued in due course.

If a service interruption (e.g. due to Order Cache failing over to its mirror or both Order Cache processes failing) occurs in the Drop Copy Gateway while it is servicing an Order Mass Status Request, the gateway will send an unsolicited Execution Report with a 'Rejected' state. It would include the MassStatusReqID (584) of the request, an ExecID (17) of ‘0’ an ExecType (150) of Order Status (I) an OrdStatus (39) of Rejected (8) and an OrdRejReason (103) of ‘10005’ – Application Unavailable. When the client receives this, he is expected toresubmit the Mass Status Request. .

If a client requests an open order book download when the service is unavailable, (e.g. both Order Cache instances are down) the request will be rejected with a business reject, with reason 4 – Application Unavailable.

In the unlikely event of a site outage disaster on the The JSE~~The JSE~~ system, all orders will be cancelled and all unicast and multicast connectivity will be unavailable until the recovery site is invoked.

5 FIX CONNECTIONS AND SESSIONS

5.1 Establishing a FIX Connection

FIX connections and sessions between the client and server are maintained as specified in the FIXT protocol.

Each client will use the assigned IP address and port to establish a TCP/IP session with the server. The client will initiate a FIX session at the start of each trading day by sending the Logon message. The client will identify itself using the Interface User ID (SenderCompID (49) field).

The server will validate the Interface User (CompID), password and IP address of the client.

System can be configured in such a way that the test request at logon is either disabled or enabled[2]. A test request will not be sent along with the logon reply if the test request switch is set to disabled. Depending on the System Configuration, the client's logon message will be responded in two ways. The JSE has configured the Drop Copy Gateway to be defaulted with the Test Request enabled upon login.

If during a logon of an Interface User ID (SenderCompID), the server receives a second connection attempt via the same TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will immediately break the TCP/IP connection with the client without sending any messages. If the server receives another connection attempt from the same SenderCompID, while a session is already established, the connection attempt will be rejected via a Reject message without breaking the existing TCP/IP connection with the client. The server will increment the next inbound message sequence number expected from the client as well as its own outbound message sequence number.

5.1.1 Test Request at logon Disabled

Once the client is authenticated, the server will respond with a Logon message. The SessionStatus (1409) of this message will be Session Active (0). If the client's Logon message included the field NewPassword (925) and the client is authenticated, the server will respond with a Logon message. The SessionStatus (1409) of this message will be Session Active (0).

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a Resend Request. Once the response/s to the Resend Request is processed by the FIX Gateway, the FIX Gateway would send a Test Request to make sure both the client and server is in sync before sending out any missed or new application messages.

The client must wait for the server's Logon before sending additional messages. The server will break the TCP/IP connection if messages are received before the exchange of Logons.

5.1.2 Test Request at logon Enabled (default)

Currently it is expected that the Client who connects to the server responds to the Test Request message sent by the server with a Heartbeat or Resend Request Message. If the client does not comply with this and sends Application Messages such as Order Mass Status Request before responding to the Test Request message, the server will not send a Reject message until the Client has responded to the Test Request message.

Immediately after the server sends the response to the Logon message, if the logon was successful, the server will send a Test Request with a specific TestRequestID (112), to identify if the Client is in sync with the server's outgoing sequence number prior to sending any (i.e. new or missed) Application Messages. If the Client replies to the Test Request with a Heartbeat message with the same TestRequestID (112) as in the Test Request, then that would imply that the client recognises the messages sent by the server and that the sequence numbers are in sync. The server would start sending any new or missed Application

[2] This is done via the process configuration CLIENT SESSION CONFIRMATION.

Messages to the Client only after receiving this Heartbeat message. Any Application Messages submitted before responding to the Test Request will be rejected with a Business Message Reject. Such a message will include a BusinessRejectReason (380) of Session not in sync (0) and an indication that the rejection was due to the client session not being in sync with the server in the Text (58) field.

The client must send additional messages to the server only after responding to the Test Request. If the client sends any Application Messages before responding to the Test Request as mentioned above, those messages will be processed by the system and are not rejected, however the Client will not receive any responses to the requests as the server has not established that the Client is in sync with the server's outgoing sequence numbers. When the Client sends a Logon message, and if the server receives a higher sequence number than expected, the server sends a Resend Request followed immediately by a Test Request. The client should respond to the Resend Request and afterwards respond to the Test Request to get back in sync with the server.

If the client ignores the Test Request because the sequence number in the message is higher than the expected sequence number, the Client is expected to send a Resend Request asking for the missed messages. After responding to the Resend Request the FIX Gateway will send another Test Request to make sure both the client and server is in sync before sending out any missed or new application messages.

If the client sends a Resend Request before the FIX Gateway sends a Test Request, then the FIX Gateway will serve the Resend Request first. After responding to the Resend Request the FIX Gateway will send a Test Request to make sure both the client and server are in sync before sending out any missed or new application messages.

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a Resend Request and once the response/s to the Resend Request is processed by the FIX Gateway, the FIX Gateway will send a Test Request to make sure both the client and server is in sync before sending out any missed or new application messages.

5.1.3 Behaviour common to both configurations

If a logon attempt fails because of an invalid Interface User ID (SenderCompID), invalid TargetCompID, invalid password or IP address, or not having the appropriate privileges to log into the server gateway, the server will break the TCP/IP connection with the client without sending a Logout or Reject. If during a logon of a SenderCompID, the server receives a second connection attempt via different TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will break the TCP/IP connection with the second connection without sending a Logout or Reject message. As the logon attempt failed, the server will not increment the next inbound message sequence number expected from the client. If the client tries to connect again with the same invalid information, the client will be disconnected again.

If a logon attempt fails because of an expired password, a locked Interface User ID (CompID) or if logins are not currently permitted, the server will send a Logout message and then break the TCP/IP connection with the client. In these scenarios the next inbound sequence number expected from the client will be incremented but the outbound sequence number will not be incremented. In this scenario the message sequence number 1 will be sent with the Logout message.

If a logon attempt fails because of a session level failure (e.g. due to invalid EncryptMethod or DefaultApplVerID…etc) the inbound sequence number and the outbound sequence number both will not be incremented**.** In this scenario the message sequence number 1 will be sent with the **Error! Reference source not found.** message.

However if a session level failure occurs due to a message sent by a client which contains a sequence number that is less than what is expected and the PossDupFlag (43) is not set to “Y”, then the server will send a Logout message and terminate the FIX connection. In this scenario the inbound sequence number will not be incremented but the outbound sequence number will be incremented.

If during a logon of a SenderCompID, the server receives a second connection attempt via the same TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will send a Reject message and then break the TCP/IP connection with the client. The server will increment the next inbound message sequence number expected from the client as well as its own outbound message sequence number.

The impact of logon failures on sequence numbers is summarised in the table below:

Reason for Logon Failure	Session status (of logout)	Inbound Sequence Number	Outbound Sequence Number
Invalid or expired password	8 (password expired)	Incremented by 1	Does not increase (defaulted to 1)
Locked/suspended/inactivated CompID	6 (account locked)	Incremented by 1	Does not increase (defaulted to 1)
Logins are not currently permitted	7 (logins are not allowed)	Incremented by 1	Does not increase (defaulted to 1)
Session level failure (e.g. due to invalid EncryptMethod or DefaultApplVerID etc)	101 (logout session level failure)	Does not increase	Does not increase (defaulted to 1)
Login sequence number is less than the expected sequence number	101 (logout session level failure)	Does not increase	Incremented by 1
Second connection attempt via same TCP/IP connection while a valid FIX session is already underway for that same SenderCompID	No Session logout. Reject message with Message Type 3 = Reject	Incremented by 1	Incremented by 1
Second connection attempt via different TCP/IP connection while a valid FIX session is already underway for that same SenderCompID	No logout or reject message	Does not increase	Does not increase

5.2 Maintaining a FIX Session

5.2.1 Message Sequence Numbers

As outlined in the FIXT protocol, the client and server will each maintain a separate and independent set of incoming and outgoing message sequence numbers. Sequence numbers will be initialized to 1 (one) at the start of the FIX session and be incremented throughout the session.

Monitoring sequence numbers will enable parties to identify and react to missed messages and to gracefully synchronize applications when reconnecting during a FIX session.

If any message sent by the client contains a sequence number that is less than what is expected and the PossDupFlag (43) is not set to “Y”, the server will send a Logout message and terminate the FIX connection. The Logout will contain the next expected sequence number in the Text (58) field.

If the server receives a message that cannot be processed (malformed message) it will not respond to that message and will not increment the sequence number maintained. In such a scenario, when the next readable message is received by the server it will detect a sequence gap between the client and server. The server will send a Resend Request to the client

requesting for messages from the sequence number the server is maintaining. If the client does not correct the malformed message to a readable one, the above event model will be repeated until there is no sequence gap.

A FIX session will not continue to the next trading day. The server will initialize its sequence numbers at the start of each day. The client is expected to employ the same logic.

5.2.2 Heartbeats

The client and server will use the Heartbeat message to exercise the communication line during periods of inactivity and to verify that the interfaces at each end are available. The heartbeat interval will be the HeartBtInt (108) specified in the client's Logon message.

The server will send a Heartbeat anytime it has not transmitted a message for the heartbeat interval. The client is expected to employ the same logic.

If the server detects inactivity for a period longer than the heartbeat interval plus a reasonable transmission time, it will send a Test Request message to force a Heartbeat from the client. If a response to the Test Request is not received by a reasonable transmission time, the server will send a Logout and break the TCP/IP connection. The client is expected to employ similar logic if inactivity is detected on the part of the server.

5.2.3 Increasing Expected Sequence Number

The client or server may use the Sequence Reset message in Gap Fill mode if it wishes to increase the expected incoming sequence number of the other party.

The client or server may also use the Sequence Reset message in Sequence Reset mode if it wishes to increase the expected incoming sequence number of the other party. The MsgSeqNum (34) in the header of such a message will be ignored. The Sequence Reset mode will only be used to recover from an emergency situation. It will not be relied upon as a regular practice.

5.3 Terminating a FIX Connection

The client is expected to terminate each FIX connection at the end of each trading day before the server shuts down. The client will terminate a connection by sending the Logout message. The server will respond with a Logout to confirm the termination. The client will then break the TCP/IP connection with the server. As recommended in the FIXT protocol, clients are advised to transmit a Test Request, to force a Heartbeat from the server, before initiating the logout process.

All open TCP/IP connections will be terminated by the server when it shuts down (a Logout will not be sent). Under exceptional circumstances the server may initiate the termination of a connection during the trading day by sending the Logout message. The server will terminate the TCP/IP connection (a Logout will not be sent) if the number of messages that are buffered for a client exceeds five times of the MAX_BUFFERED_COUNT<500> specified..

If, during the exchange of Logout messages, the client or server detects a sequence gap, it will send a Resend Request.

5.4 Re-Establishing a FIX Session

If a FIX connection is terminated during the trading day it may be re-established via an exchange of Logon messages. Once the FIX session is re-established, the message sequence numbers will continue from the last message successfully transmitted prior to the termination.

5.4.1 Test Request at logon Disabled

Once the client is authenticated, the server will respond with a Logon message. The SessionStatus (1409) of this message will be Session Active (0). If the client's Logon message

included the field NewPassword (925) and the client is authenticated, the SessionStatus (1409) of the Logon sent by the server will be Session Active (0).

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a Resend Request and once the response/s to the Resend Request is processed by the FIX Gateway, the FIX Gateway would send a Test Request to make sure both the client and server is in sync before sending out any missed or new application messages.

The client must wait for the server's Logon before sending additional messages. If additional messages are received from the client before the exchange of Logon messages, the TCP/IP connection with the client will be disconnected.

5.4.2 Test Request at logon Enabled (default)

Once the client is authenticated, the server will respond with a Logon message, followed by a Test Request. The Logon message will confirm the logon status and the Test Request's purpose is to sync the Sequence numbers before sending any Missed Messages if any. The SessionStatus (1409) of this message will be Session Active (0). If the client's Logon message included the field NewPassword (925) and the client is authenticated, the SessionStatus (1409) of the Logon sent by the server will be Session Active (0).

If the client responds to the Test Request with a Heartbeat message containing the appropriate Test Request ID and message sequence number, the server can start transmitting the missed messages or new messages in the Gateway. The client would not receive any responses to application messages sent until sequence numbers are synchronized by responding to the Test Request via Heartbeat or Resend Request message. If the client does not respond to the Test Request during the heartbeat interval, the gateway will disconnect the client.

If the client ignores the Test Request because the sequence number in the message is higher than the expected sequence number, the Client is expected to send a Resend Request asking for the missed messages. After responding to the Resend Request the FIX Gateway would send another Test Request to make sure both the client and server is in sync before sending out any missed or new application messages.

If the client sends a Resend Request before the FIX Gateway send a Test Request, then the FIX Gateway will serve the Resend Request first. After responding to the Resend Request the FIX Gateway would send a Test Request to make sure both the client and server are in sync before sending out any missed or new application messages.

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a Resend Request and once the response/s to the Resend Request is processed by the FIX Gateway, the FIX Gateway would send a Test Request to make sure both the client and server are in sync before sending out any missed or new application messages.

Once the FIX session is re-established successfully, the message sequence numbers will continue from the last message successfully transmitted prior to the termination.

5.4.3 Resetting Sequence Numbers: Starting a New FIX Session

5.4.3.1 Reset Initiated by the Client

If the client requires both parties to initialize (i.e. reset to 1) sequence numbers, it may use the ResetSeqNumFlag (141) field of the Logon message. The server will respond with a Logon with the ResetSeqNumFlag (141) field set to "Y" to confirm the initialization of sequence numbers.

A client may also manually request JSE Market Operations that it would like the server to initialize its sequence numbers prior to the client's next login attempt.

These features are intended to help a client manage an emergency situation. Initializing sequence numbers on a re-login will not be relied upon as a regular practice.

5.4.3.2 **Reset Initiated by the Server**

The System has been designed with fault tolerance and disaster recovery technology that will ensure that the server retains its incoming and outgoing message sequence numbers for each client in the unlikely event of an outage.

However, clients are also required to support a manual request by JSE Market Operations to initialize sequence numbers prior to the next login attempt.

6 RECOVERY

6.1 Resend Requests

The client may use the Resend Request message to recover any lost messages. As outlined in the FIXT protocol, this message may be used in one of three modes:

(i) To request a single message. The BeginSeqNo (7) and EndSeqNo (16) should be the same.

(ii) To request a specific range of messages. The BeginSeqNo (7) should be the first message of the range and the EndSeqNo (16) should be the last of the range.

(iii) To request all messages after a particular message. The BeginSeqNo (7) should be the sequence number immediately after that of the last processed message and the EndSeqNo (16) should be zero (0).

The server caches a certain number of messages RESEND_CACHE_SIZE <2000> transmitted to each Interface User (CompID). Clients are unable to use a Resend Request to recover messages not in the server's cache.

6.2 Possible Duplicates

The server handles possible duplicates according to the FIX protocol. The client and server will use the PossDupFlag (43) field to indicate that a message may have been previously transmitted with the same MsgSeqNum (34).

6.3 Possible Resends

The server does not handle possible resends for client-initiated messages and ignores the value in the PossResend (97) field of such messages.

The server may, in the circumstances outlined in Section 6.4 use the PossResend (97) field to indicate that an Execution Report may have already been sent under a different MsgSeqNum (34). The client will validate the ExecID (17) of such a message against those of Execution Reports already received during the current trading day.

If an Execution Report with same ExecID (17) had been processed, the resent message should be ignored. If the same ExecID (17) had not been processed, the Execution Report should be processed.

6.4 Transmission of Missed Messages

The Execution Reports generated during a period when a client is disconnected from the server will be sent to the client when it next reconnects. In the unlikely event the disconnection was due to an outage of the server, all such messages will include a PossResend (97) of "Y.

This feature is intended to help a client manage an emergency situation and it should not be relied upon as a regular practice.

7 MESSAGE FORMATS

This section provides details on the header and trailer, the seven administrative messages and three application messages utilized by the server. Client-initiated messages not included in this section are rejected by the server via a Reject or Business Message Reject.

All fields are encoded using printable ASCII. Any message not included in this section will be rejected by the server for this service.

7.1 Supported Message Types

7.1.1 Administrative Messages

All administrative messages may be initiated by either the client or the server.

Message	MsgType	Usage
Logon	A	Allows the client and server to establish a FIX session.
Logout	5	Allows the client and server to terminate a FIX session.
Heartbeat	0	Allows the client and server to exercise the communication line during periods of inactivity and verify that the interfaces at each end are available.
Test Request	1	Allows the client or server to request a response from the other party if inactivity is detected.
Resend Request	2	Allows for the recovery of messages lost during a malfunction of the communications layers.
Reject	3	Used to reject a message that does not comply with FIXT.
Sequence Reset	4	Allows the client or server to increase the expected incoming sequence number of the other party.

7.1.2 Application Messages

7.1.2.1 Client-Initiated

Message	MsgType	Usage
Order Mass Status Request	AF	Allows the client to request the status of all active orders for a particular Trader and Trader Group combination.

7.1.2.2 **Server-Initiated**

Message	MsgType	Usage
Execution Report	8	Indicates one of the following: (i) Order accepted (ii) Order rejected (iii) Order or quote (single or mass or privately negotiated) executed (iv) Order expired (v) Order cancelled (vi) Order cancel/replaced (vii) Quote in a privately negotiated RFQ conversation is accepted. (viii) Quote in a privately negotiated RFQ conversation is rejected. (ix) Quote in a privately negotiated RFQ conversation is cancelled. (x) Quote in a privately negotiated RFQ conversation is expired. (xi) Trade cancellation or correction (xii) Order status (xiii) Order mass status request rejected
Business Message Reject	J	Indicates that an application message could not be processed.

7.2 Message Header and Trailer

7.2.1 Message Header

Tag	Field Name	Req	Description
8	BeginString	Y	FIXT.1.1
9	BodyLength	Y	Number of characters after this field up to and including the delimiter immediately preceding the CheckSum.
35	MsgType	Y	Message type.
49	SenderCompID	Y	Interface User (CompID) of the party sending the message.
56	TargetCompID	Y	Interface User (CompID) of the party the message is sent to.
115	OnBehalfOf CompID	N	Required for server-initiated application messages. This will be the Interface User (CompID) of the connection that originated the order referenced in the message being drop copied.
34	MsgSeqNum	Y	Sequence number of the message.
43	PossDupFlag	N	Whether the message was previously transmitted under the same MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N). **Value** **Meaning** Y Possible Duplicate N Original Transmission
97	PossResend	N	Whether the message was previously transmitted under a different MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N). **Value** **Meaning** Y Possible Resend N Original Transmission
52	SendingTime	N	Time the message was transmitted. Not required for incoming messages sent by the clients (even if sent by a client, no validation will be done). Required for outgoing messages sent by the server.
122	OrigSendingTime	N	Time the message was originally transmitted. If the original time is not available, this will be the same value as SendingTime (52). Required if PossDupFlag (43) is Possible Duplicate (Y).
1128	ApplVerID	N	Version of FIX used in the message. Required if the message is generated by the server. **Value** **Meaning** 9 FIX50SP2

7.2.2 Message Trailer

Tag	Field Name	Req	Description
10	CheckSum	Y	

7.3 Administrative Messages

7.3.1 Logon

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	A = Logon
Message Body			
98	EncryptMethod	Y	Method of encryption. **Value** **Meaning** 0 None
108	HeartBtInt	Y	Indicates the heartbeat interval in seconds.
141	ResetSeqNum Flag	N	Indicates whether the client and server will reset sequence numbers. Absence of this field is interpreted as Do Not Reset Sequence Numbers (N). **Value** **Meaning** Y Reset Sequence Numbers N Do Not Reset Sequence Numbers
554	Password	N	Password assigned to the Interface User (CompID). Required if the message is generated by the client.
925	NewPassword	N	New password for the Interface User (CompID).
1409	SessionStatus	N	Status of the FIX session. Required if the message is generated by the server. **Value** **Meaning** 0 Session Active 2 Password Due to Expire.
1137	DefaultApplVerID	Y	Default version of FIX messages used in this session which will be validated by the server. **Value** **Meaning** 9 FIX50SP2
Standard Trailer			

7.3.2 Logout

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	5 = Logout
Message Body			
1409	SessionStatus	N	Status of the FIX session. Required if the message is generated by the server. **Value** – **Meaning** 4 – Session logout complete 6 – Account locked 7 – Logons are not allowed at this time 8 – Password expired 100 – Other 101 – Logout due to session level failure 102 – Logout by market operations
58	Text	N	This field will contain the next expected sequence number as well as the received sequence number if the server terminated the connection after receiving a sequence number that was less than what was expected. In other cases the field will contain the reason for the logout.
Standard Trailer			

7.3.3 Heartbeat

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	0 = Heartbeat
Message Body			
112	TestReqID	N	Required if the heartbeat is a response to a Test Request. The value in this field will echo the TestReqID (112) received in the Test Request.
Standard Trailer			

7.3.4 Test Request

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	1 = Test Request
Message Body			
112	TestReqID	Y	Identifier for the request.
Standard Trailer			

7.3.5 Resend Request

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	2 = Resend Request
Message Body			
7	BeginSeqNo	Y	Sequence number of first message in range.
16	EndSeqNo	Y	Sequence number of last message in range.
Standard Trailer			

7.3.6 Reject

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	3 = Reject
Message Body			
45	RefSeqNum	Y	MsgSeqNum (34) of the rejected message.
372	RefMsgType	N	MsgType (35) of the rejected message.
371	RefTagID	N	If a message is rejected due to an issue with a particular field its tag number will be indicated.
373	SessionReject Reason	N	Code specifying the reason for the reject. Please refer to Section **Error! Reference source not found.** for a list of reject codes.
58	Text	N	Text specifying the reason for the rejection.
Standard Trailer			

7.3.7 Sequence Reset

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	4 = Sequence Reset
Message Body			
36	NewSeqNo	Y	Sequence number of the next message to be transmitted.
123	GapFillFlag	N	Mode in which the message is being used. Absence of this field is interpreted as Sequence Reset (N). **Value** – **Meaning** Y – Gap Fill N – Sequence Reset
Standard Trailer			

7.4 Application Messages (Client-Initiated)

7.4.1 Order Mass Status Request

Tag	Field Name		Req	Description
Standard Header				
35	MsgType		Y	AF = Order Mass Status Request
Message Body				
584	MassStatus ReqID		Y	Client specified identifier of the mass status request.
585	MassStatus ReqType		Y	Type of mass status request. **Value** – **Meaning** 8 – All open orders of specified PartyID 1 – All open orders for a specific instrument and PartyID combination 100 – All open orders for a specific segment and PartyID combination
453	NoPartyIDs		Y	JSE Market requires 3 blocks to represent the following: 1. Trader 2. Trader Group 3. Executing Firm The Trader and Trader Group will be concatenated in the System to uniquely identify the Trader ID. Possible values for this field are: 1 (to request open orders of the Firm) 2 (to request open orders of a particular trader, specifying the Trader and Trader group combination) Therefore Order Mass Status request can be performed for a specified Firm OR Trader and Trader Group combination.
➡	448	PartyID	Y	Identifier of the user. (Trader and trader group combination
➡	447	PartyID Source	Y	**Value** – **Meaning** D – Proprietary/Custom Code
➡	452	Party Role	Y	Role of the PartyID (448). **Value** – **Meaning** 1 – Executing Firm 53 – Trader 76 – Trader Group
48	SecurityID		N	Identifier of the instrument the order mass status is requested for. This will be the Instrument ID Required if MassStatus ReqType (585) is '1'

22	SecurityIDSource	N	Identifier of the source of the SecurityID (48) value the order mass status is requested for. Required if SecurityID (48) is specified. **Value** **Meaning** 8 Exchange Symbol
1300	MarketSegmentID	N	Identifier of the segment the order mass status is requested for. Required if MassStatus ReqType (585) is '100'
Standard Trailer			

7.5 Application Messages (Server-Initiated)

7.5.1 Execution Report

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	8 = Execution Report
Message Body			
1180	ApplID	Y	Identity of the partition that generated the message.
17	ExecID	Y	Server specified identifier of the message. Will be “0” if ExecType (150) is Order Status (I).
11	ClOrdID	N	Client specified identifier of the order. Please note that when an Order Mass Status Request is rejected in its entirety, the ClOrdID (11) will not be present as the rejection is not related to a specific order.
526	SecondaryClOrdID	N	Assigned by the party which originates the order. Can be used to include an additional unique identifier.
41	OrigClOrdID	N	OrigClOrdID (41), if it was submitted with the order cancel or cancel/replace request.
37	OrderID	N	Server specified identifier of the order. This will be a 62 base encoded value in ASCII format. By converting this to binary, this can be mapped with MITCH Order ID (refer to Section 3.9). Please note that when an Order Mass Status Request is rejected in its entirety, the OrdID (37) will not be present as the rejection is not related to a specific order.
584	MassStatus ReqID	N	Client specified identifier of the mass status request. Required if the message is sent in response to such a request.
912	LastRpt Requested	N	Indicates the last message sent in response to a mass order status request. This will be set for the last message sent for each partition. **Value** **Meaning** Y Last Message
442	MultiLeg ReportingType	N	Type of trade. Values disseminated in this field when ExecType (150) is not 'Trade' (F) should be ignored. **Value** **Meaning** 1 Trade of Single Instrument 2 Leg Trade of a Multi-Leg Instrument Trade 3 Trade of Multi-Leg Instrument
198	Secondary OrderID	N	Server specified identifier of the order or quote side for the multi-legged instrument (i.e. strategy instrument). Required if MultiLeg ReportingType (442) is Leg Trade of a Multi-Leg Instrument Trade (2).

<table>
<tr><td>150</td><td>ExecType</td><td>Y</td><td>Reason the execution report was generated.
Value Meaning
0 New
4 Cancelled
5 Replaced
8 Rejected
9 Suspended
C Expired
D Restated
F Trade
G Trade Correct
H Trade Cancel
I Order Status
L Triggered</td></tr>
<tr><td>880</td><td>TrdMatchID</td><td>N</td><td>Identifier of the trade. This will be a 62 base encoded value in ASCII format. Required if ExecType (150) is Trade (F), Trade Correct (G) or Trade Cancel (H).</td></tr>
<tr><td>19</td><td>ExecRefID</td><td>N</td><td>Reference to the execution being cancelled or corrected. Required if ExecType (150) is Trade Cancel (H) or Trade Correct (G).</td></tr>
<tr><td>378</td><td>Exec Restatement Reason</td><td>N</td><td>This is used to indicate if an order was Cancelled or Cancel/Replaced by Market Operations. Required if ExecType (150) is Cancelled (4) or Replaced (5).
Value Meaning
8 Market Option</td></tr>
<tr><td>336</td><td>Trading SessionID</td><td>N</td><td>Session the order is valid for.
Value Meaning
a Closing Price Cross
b Good for Volume Auction Uncross(GDX)
This field will only be used to identify an order that may be executed during the Closing Price Cross session.</td></tr>
<tr><td>39</td><td>OrdStatus</td><td>Y</td><td>Current status of the order.
Value Meaning
0 New
1 Partially Filled
2 Filled
4 Cancelled
8 Rejected
9 Suspended
C Expired</td></tr>
</table>

636	Working Indicator	N	. Whether the order is currently being worked (elected) or not. **Value** **Meaning** N Order is Not in a Working State (Order is accepted but in an unelected state/parked queue.) Y Order is Being Worked (Order has been elected and has been added to the normal order book).
103	OrdRejReason	N	Code specifying the reason for the reject. Please refer to Section **Error! Reference source not found.** for a list of reject codes. Required if ExecType (150) is Rejected (8).
58	Text	N	Text specifying the reason for the rejection or expiration
32	LastQty	N	Quantity executed in this fill. Required if ExecType (150) is Trade (F) or Trade Correct (G).
31	LastPx	N	Price of this fill. Required if ExecType (150) is Trade (F) or Trade Correct (G).
669	LastParPx	N	Price of this fill expressed as percentage of par. Required if LastPx (31) is specified and the trade is for a fixed income instrument quoted on discount rate or yield.
236	Yield	N	Implied yield of this fill. Required if LastPx (31) is specified and the trade is for a fixed income instrument quoted on percentage of par.
159	AccruedInterest Amt	N	Accrued Interest for a unit trade if executed on the current trading day. Multiplying this by the LastQty (32), if available, gives the Total accrued interest. May only apply for a Regular Coupon Bond or TIPS.
151	LeavesQty	N	Quantity available for further execution. It is the remaining quantity of the order. Will be “0” if OrdStatus (39) is Filled (2), Cancelled (4), Rejected (8) or Expired (C). Will not be stamped if RFQ quote is rejected.
14	CumQty	Y	Total cumulative quantity filled.
48	SecurityID	Y	Unique Identifier of the instrument. This will be the Instrument ID
1917	CoverPrice	N	The price that was the best available in the quote negotiation. Required if the execution report is sent to other market makers denoting a successful execution during a negotiation via RFQs with another market maker. Cover price will not be stamped if the ER is sent for the side which did not receive the execution in a dual sided quote (i.e. the dual sided quote did not receive the execution at all or one side receives the execution during the negotiation. In both cases the side which was not involved in the negotiation process will not have cover price stamped)

32021	ParPx	N	Converted clean price of an order's limit price. If computed, it will be on the Price (44) of an order belonging to a fixed income instrument quoted on discount rate or yield.
30005	ConvertedYield	N	Converted yield value of an order's limit price. If computed, it will be on the Price (44) of an order belonging to a fixed income instrument quoted on percentage-of-par (i.e. on price).
22	SecurityIDSource	Y	Identifier of the source of the SecurityID (48) value. **Value** **Meaning** 8 Exchange Symbol
30001	OrderBook	Y	**Value** **Meaning** 1 Normal 11 Negotiated Trades
1	Account	Y	Client Account information. This is the Client Account of the firm who is sending the order.
Component Block <Trading Party>		Y	Identifier of the trading party.

<table>
<tr><th>Tag</th><th>Field Name</th><th>Req</th><th>Description</th></tr>
<tr><td>40</td><td>OrdType</td><td>Y</td><td>Type of the order.
Value Meaning
1 Market
2 Limit
3 Stop
4 Stop Limit
K Market to Limit
J Market If Touched
P Pegged
R Pegged Limit</td></tr>
<tr><td>59</td><td>TimeInForce</td><td>N</td><td>Time qualifier of the order. Absence of this field is interpreted as Day (0).
Value Meaning
0 Day
1 Good Till Cancel (GTC)
2 At the Opening (OPG)
3 Immediate or Cancel (IOC)
4 Fill or Kill (FOK)
6 Good Till Date (GTD)
7 At the Close (ATC)
8 Good for Intra-Day Auction (GFX)
9 Good for Auction (GFA)
(GTT is specified using 6 in this field and specifying an Expire Time.)</td></tr>
<tr><td>126</td><td>ExpireTime</td><td>N</td><td>Time the order expires which must be a time during the current trading day. Required if TimeInForce (59) is GTD (6) and ExpireDate (432) is not specified.
If the ExpireTime (126) and ExpireDate (432) are specified, the message will be rejected.</td></tr>
<tr><td>432</td><td>ExpireDate</td><td>N</td><td>Date the order expires. Required if TimeInForce (59) is GTD (6) and ExpireTime (126) is not specified.</td></tr>
<tr><td>54</td><td>Side</td><td>Y</td><td>Side of the order or quote that was executed. Will not be stamped if RFQ quote is rejected
Value Meaning
1 Buy
2 Sell</td></tr>
</table>

<table>
<tr><td>38</td><td>OrderQty</td><td>N</td><td>Total order quantity. In the case of a quote, the executed side's size submitted with the last quote update. In the case of RFQ quotes, this will not be stamped if sent for a rejection
Order Quantity = Leaves Quantity + Cumulative Executed Quantity</td></tr>
<tr><td>1138</td><td>DisplayQty</td><td>N</td><td>Quantity currently displayed in the order book. This is the Visible Size. It is equal to zero for Hidden Pegged/Pegged Limit orders and equal to the Order quantity for normal order. If the field DisplayMethod (1084) has a value of 4, DisplayQty will be zero</td></tr>
<tr><td>1084</td><td>DisplayMethod</td><td>N</td><td>Whether the order is a Hidden Pegged/Pegged Limit order.
Required for hidden orders.
Value Meaning
4 Undisclosed (Hidden Pegged/Pegged Limit Order)</td></tr>
<tr><td>110</td><td>MinQty</td><td>N</td><td>Minimum Execution Size that needs to be specified for a Hidden Pegged or Pegged Limit Order which must be greater than or equal to Minimum Reserve Size.</td></tr>
<tr><td>c44</td><td>Price</td><td>N</td><td>Limit price. Required if OrderType (40) is Limit (2) or Stop Limit (4). In the case of a quote, the executed side's price submitted with the last quote update. In the case of RFQ quotes, this will not be stamped if sent for a rejection</td></tr>
<tr><td>694</td><td>QuoteRespType</td><td>N</td><td>Value Meaning
1 Hit/Lift
3 Expired
4 Cover
5 Done Away
7 End Trade
8 Timed Out
9 Tied
10 Tied Cover
12 Contra Side</td></tr>
<tr><td>300</td><td>QuoteRejectReason</td><td>N</td><td>Conditionally required if QuoteAckStatus(1865) is Rejected(2).</td></tr>
<tr><td>99</td><td>StopPx</td><td>N</td><td>Stop price/Trigger price. Required if OrderType (40) is Stop (3) or Stop Limit (4) or Market If Touched (J).</td></tr>
<tr><td>211</td><td>PegOffsetValue</td><td>N</td><td>Trailing Offset added to trailing stop/stop limit orders. Only positive values are allowed with the value zero.
This is applicable only for the derivatives/bonds markets.</td></tr>
</table>

1091	PreTrade Anonymity	N	Whether the order is anonymous or named. Absence of this field is interpreted as Anonymous (Y). **Value** **Meaning** Y Anonymous N Named
528	OrderCapacity	Y	Capacity of the order. **Value** **Meaning** A Agency P Principal
60	TransactTime	Y	Time the transaction represented by the Execution Report occurred.
27000	IsMarketOpsRequest	N	Identifies whether an order was submitted on behalf of a client by Market Operations. **Value** **Meaning** 1 Yes
18	ExecInst	N	**Value** **Meaning** u Include in Volume Auction Uncross v Exclude Hidden Limit Orders
30006	RFQID	N	Server specified identifier of a Private RFQ
921	StartCash	N	The all-in-price of an EFP trade.
194	LastSpotRate	N	The spot rate used in an EFP trade or repo rate used by a responding market maker for a repo trade.
829	TrdSubType	N	The trade type of an RFQ trade.
922	End Cash	N	2nd leg dirty price of a repo trade.
916	Start Date	N	Starting date of repo trade
917	End Date	N	Date of settlement for the 2nd leg of the repo trade.
32022	LastOptPx	N	Converted price of the executed volatility of the options instrument.
1188	Volatility	N	Converted volatility of the executed price of the options instrument.
548	Cross ID	N	The unique ID of the cross order. Only populated for execution report messages generated for Internal cross orders. The value submitted with the New Order Cross Message will be populated.
549	Cross Type	N	The type of the cross order. Only populated for execution report messages generated Internal cross orders. The value submitted with the New Order Cross Message ~~message~~ will be populated. Value Meaning 5 Internal Cross 50 Internal Cross (Price Adjustable)

1094	PegPriceType	N	Required if the Order Type is Pegged or Pegged Limit. 2 = Mid-Price Peg 5 = Primary Peg (Buy at Bid, Sell at Offer)
6	AvgPx	N	Calculated average price of all fills on this order. For Fixed Income trades AvgPx is always expressed as percent-of-par. I.e., AvgPx will contain an average of percent-of-par values (see LastParPx (669)) for issues traded in Yield, Spread or Discount.
818	SecondaryTrade ReportID	N	Free-text field to pass instructions relating to the trade to back-office users or Clearing Members.
278	MDEntryID	Y	Server specified public order identifier of the order.
Standard Trailer			

7.5.2 Business Message Reject

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	j = Business Message Reject
Message Body			
45	RefSeqNum	Y	MsgSeqNum (34) of the rejected message.
372	RefMsgType	Y	MsgType (35) of the rejected message.
371	RefTagID	N	If a message is rejected due to an issue with a particular field its tag number will be indicated.
380	BusinessReject Reason	Y	Code specifying the reason for the reject. Please refer to Section 8.2 for a list of reject codes.
58	Text	N	Text specifying the reason for the rejection.
Standard Trailer			

7.5.3 Components of Application Messages

7.5.3.1 Trading Party

Tag	Field Name		Req	Description
453	NoPartyIDs		Y	Equity and derivatives/bond markets requires 6 blocks to represent the following: 1. Trader 2. Trader Group 3. Firm 4. Contra Firm 5. Contra Trader 6. Market Makers The JSE Trader ID and Trader Group will be concatenated in the System to uniquely identify the Trader. The value in this field will be “2” or “3” when generated for orders. 2 – Trader and Trader Group combination. 3 – Executing Firm, Trader and Trader Group combination. On order entry it is not required to specify the Firm. On the FIX execution reports generated, the Firm ID will be returned through the party block. When submitting a Quote Request, it is required to specify the Trader (53), Trader group (76) and the Market Maker Firm IDs (66). When an RFQ Execution Report is sent to Requester, User ID of the Requester as Trader (53), Trader group (76) and the Market Maker Firm ID as the Contra Firm (17) will be specified. When a Quote Cancellation (via Service Desk) is communicated to Market Maker User ID of the Requester as Trader (53) and Trader group (76) will be specified.
➡	448	PartyID	Y	Identifier of the party.
➡	447	PartyID Source	Y	**Value** – **Meaning** D – Proprietary/Custom Code
➡	452	Party Role	Y	Role of the specified PartyID (448). **Value** – **Meaning** 53 – Trader 76 – Trader Group 1 – Executing Firm 17 – Contra Firm 37 – Contra Trader 66 – Market Makers (Firm IDs)

8 REJECT CODES

Please refer to Volume 10 – JSE Reject Codes Specification for the list of reject codes and meanings specific to the JSE and NSX markets.

8.1 Functional and Implementation Limitations

All the FIX gateways (FIX Trading, FIX Drop Copy and FIX Post Trade) currently use a common library. The system hence accepts all FIX messages defined for all three gateways, and cannot distinguish between them per gateway.

It will validate the incoming messages in the following sequence:

(a) The system initially does a FIX library level validation
(b) The system does a validation for required fields
(c) The system finally does the Gateway level validation

Hence;

(a) If a message is sent which does not comply with the specific gateway being used (but is defined in a different FIX gateway), it will validate the required fields. If any of the required tags are missing, it will give out a session reject with message "Required tag missing').
(b) If a message is sent which does not comply with the specific gateway being used (but is defined in a different FIX gateway), it will validate the required fields. If all required fields are available, a gateway validation gives out a business reject message "Unsupported Message Type".
(c) If a message is sent which does not comply with any of the FIX gateways used it will then give out a session reject message "Invalid Msg Type".

When an Order Mass Status Request is rejected at its entirety, an Execution Report is generated but it does not carry a client order id as the rejection is not related to a specific order. Hence there is an exception to the fact that tag 11 is required in the Execution Report.

If an undefined tag is sent along with any of the Administrative messages, then the system will ignore the undefined tags and process the rest of the message. This is a limitation exists in the FIX library**.**

The maximum length of the PartyID (448) field is 17. The value will be truncated to length 17 prior to the “user” validation. The system will accept the order if there is a corresponding trader group for the value after the truncation. Otherwise, the order will be rejected with reject reason "Unknown user (OwnerID)".

Example: Order with PartyID FT05TR011123456789 is entered to the system. (PartyID FT05TR011123456789 has more than 17 characters and not a configured user in the system). However, the system will truncate the trader group to 17 characters; FT05TR01112. If there is a corresponding trader for FT05TR01112345678 the system will accept the order. Otherwise, the system will reject the order stating “Unknown User”.

8.2 Business Message Reject

Business Reject Reason	Meaning
0	Other
2	Unknown Symbol
3	Unsupported message type. This reject reason is sent when the received message is not defined as a valid message for the Drop Copy Gateway.
4	Application not available
5	Conditionally required field missing
30	Session not in sync