Johannesburg Stock Exchange

Trading and Information Solution

JSE Specification Document

Volume 06 – Market Data Gateway (FAST - UDP)

Version	3.01
Release Date	4 August 2016
Number of Pages	77 (Including Cover Page)

1 DOCUMENT CONTROL

1.1 Table of Contents

1 DOCUMENT CONTROL 2

1.1 Table of Contents 2
1.2 Document Information 4
1.3 Revision History 4
1.4 Open Issues 4
1.5 Circulation ..... **Error! Bookmark not defined.**
1.6 References 4
1.7 Definitions, Acronyms and Abbreviations 4

2 OVERVIEW 8

3 FAST GATEWAY SERVICE DESCRIPTION 12

3.1 System Architecture 12
3.1.1 Real-Time Channel 12
3.1.2 Recovery Channel 13
3.1.3 Replay Channel 13
3.2 Message Overview 15
3.3 Overview of a Trading Day 17
3.3.1 Trading on the Order Book 17
3.3.2 Trade Reporting 17
3.3.3 List of Instruments 18
3.3.4 Ex-Marker and Annotation Information 18
3.3.5 Trading Status 18
3.3.6 Trading Halt 18
3.3.7 Pause 19
3.3.8 Trading Suspension 19
3.3.9 Intra-Day Trading Session Updates 19
3.4 Order Book Management (Price Depth) 21
3.4.1 Incremental Refresh 21
3.4.2 Snapshot 22
3.4.3 Market Orders 22
3.5 Time and Sales 22
3.5.1 Off Book Trades 22
3.5.2 Auction Trades 23
3.5.3 Trade Cancellation and Corrections 23
3.6 Indicative Auction Information 23
3.7 Statistics 24
3.7.1 Incremental Update 24
3.7.2 Snapshot 25
3.8 Quotation Conventions 25
3.9 Market Operations Announcements 25

4 CONNECTIVITY 26

4.1 Transmission Standards 26
4.1.1 Multicast Channels 26
4.1.2 Point-to-Point Channels 26
4.2 Application IDs (ApplIDs) 26
4.2.1 Server 26
4.2.2 Clients 26
4.3 Production IP Addresses and Ports 27

5 RECOVERY 28

5.1 Recipient Failures 28
5.1.1 Recovery Channel 28
5.1.2 Replay Channel 32

5.2 Failures at the JSE 34
5.2.1 Snapshots on the Real-Time Channel 34
5.2.2 Resetting Sequence Numbers 35

6 MESSAGE FORMATS AND TEMPLATES 36

6.1 Variations from the FIX Protocol 36
6.2 Administrative Messages 38
6.2.1 Logon 38
6.2.2 Logout 39
6.2.3 Heartbeat 40
6.3 Application Messages (Client-Initiated) 41
6.3.1 Security Definition Request 41
6.3.2 Market Data Request 42
6.3.3 Application Message Request 45
6.4 Application Messages (Server-Initiated) 46
6.4.1 Security Definition 46
6.4.2 Security Status 49
6.4.3 Market Data Snapshot (Full Refresh) 52
6.4.4 Market Data Incremental Refresh 59
6.4.5 News 66
6.4.6 Market Data Request Reject 68
6.4.7 Business Message Reject 69
6.4.8 Application Message Request Ack 70
6.4.9 Application Message Report 70

7 INSTRUMENT CATEGORIES 72

7.1 Segment 72

8 TRADING HALT REASON CODES 73

9 REJECT CODES 74

9.1 Market Data Request Reject 74
9.2 Business Message Reject 74

10 VALID TRADE TYPES 75

10.1 TrdSubType 75

11 CORPORATE ACTION INFORMATION 76

11.1 Valid Ex-Markers 76
11.2 Valid Annotations 76

12 MARKET DATA SERVICE DIAGRAMS ..... ERROR! BOOKMARK NOT DEFINED.

12.1 Market Data Services for JSE ..... **Error! Bookmark not defined.**
12.2 Market Data Services for NSX ..... **Error! Bookmark not defined.**

1.2 Document Information

Drafted By	Trading and Market Services: TMS Trading
Status	Final
Version	3.01
Release Date	4 August 2016

1.3 Revision History

Date	Version	Description
08 July 2011	1.00	Initial Draft
30 November 2011	1.01	JSE Specification updates – NSX ITCH and LVL2 Snapshot change request updates to follow
13 March 2012	1.02	Updates to include the new Futures Closeout Auction changes
5 July 2013	2.00	Functionality updates related to the 2013 product upgrade
20 October 2014	2.01	Updates made to Section 3.4.1.3 against the '-1' value being sent at end of market when the orders on which the closing price was determined are expired.
29 February 2016	3.00	Integrated Trading and Clearing Project changes - DRAFT
4 August 2016	3.01	Update to the description of Cancelling a Retransmission Request

1.4 References

FAST 1.1 Session Control Protocol Specification
FIX 5.0 (Service Pack 2) Specification
FIXT 1.1 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: ITACTradingAPI@jse.co.za

1.6 Definitions, Acronyms and Abbreviations

Automated Trades (ATs)	On Book trades executed during continuous trading.
Auction Call	The trading session immediately prior to an auction (i.e. opening, re-opening or closing). During this session orders are accumulated for execution in the auction and information on the indicative auction price is disseminated at a regular interval.
Client	A Recipient connected to the Recovery or Replay channel of the market data feed.
CPP Session	Closing Price Publication is the session where the Closing Price is calculated and published to Markets
CPX Session	Closing Price Cross is the session where automatic trading can occur at the Closing Price calculated during the CPP session
FAST	The JSE implementation will be based on Version 1.1 of the Session Control Protocol (FAST SCP 1.1) of the FIX Adapted for STreaming (FAST) Protocol specification. FAST is a binary encoding method for message oriented data streams. The encoding method reduces the size of data streams by removing redundant data, thus leveraging data affinities of a stream. The remaining data in the stream is then serialized with respect to a control structure (a template) through binary encoding in the template.
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.
FTP	File Transfer Protocol
FIXT	Version 1.1 of the Financial Information Exchange Session Protocol.
Indicative Auction Information	The Indicative Auction Price (if any) and the Indicative Auction Volume (if any) at the Indicative Auction Price
JSE	Johannesburg Stock Exchange.
MBO	Market By Order (i.e. order depth), all the orders in the book will be disseminated on the feed. Time priority will be considered. Therefore the orders will be individually disseminated in terms of time priority.

MBP	Market By Price (i.e. price depth), the aggregated volume of orders in the book based for each price point will be disseminated on the feed.
NSX	Namibian Stock Exchange
Orders	Executable order in the order book.
Off Book Trade	A trade negotiated outside the System yet reported to the System, in accordance with the JSE rules and directives.
On Book Trade	An Automatic Trade which is a trade automatically executed in the System which can either be an Automated Trade or an Uncrossing Trade.
Recipient	A subscriber to the Real-Time multicast (UDP) channel of the market data feed who connects to the replay and recovery channels for recovery.
Repeating Group	A set of related attributes which occur more than once within a field of a message
Server	The FAST market data gateway at the JSE for the JSE and NSX markets.
SS	Snapshot
Trade Reporting	The reporting of an Off Book trade.
TCP/IP	Transmission Control Protocol is a connection-oriented protocol, which means that it requires handshaking to set up end-to-end communications. Once a connection is set up user data may be sent bi-directionally over the connection. • *Reliable* – TCP manages message acknowledgment, retransmission and timeout. Multiple attempts to deliver the message are made. If it gets lost along the way, the server will re-request the lost part. In TCP, there's either no missing data, or, in case of multiple timeouts, the connection is dropped. • *Ordered* – if two messages are sent over a connection in sequence, the first message will reach the receiving application first. When data segments arrive in the wrong order, TCP buffers the out-of-order data until all data can be properly re-ordered and delivered to the application. • *Heavyweight* – TCP requires three packets to set up a socket connection, before any user data can be sent. TCP handles reliability and congestion control. • *Streaming* – Data is read as a byte stream, no distinguishing indications are transmitted to signal message (segment) boundaries.

UDP	UDP is a simpler message-based connectionless protocol. Connectionless protocols do not set up a dedicated end-to-end connection. Communication is achieved by transmitting information in one direction from source to destination without verifying the readiness or state of the receiver. • *Unreliable* – When a message is sent, it cannot be known if it will reach its destination; it could get lost along the way. There is no concept of acknowledgment, retransmission or timeout. • *Not ordered* – If two messages are sent to the same recipient, the order in which they arrive cannot be predicted. • *Lightweight* – There is no ordering of messages, no tracking connections, etc. It is a small transport layer designed on top of IP. • *Datagrams* – Packets are sent individually and are checked for integrity only if they arrive. Packets have definite boundaries which are honored upon receipt, meaning a read operation at the receiver socket will yield an entire message as it was originally sent. *No congestion control* - UDP itself does not avoid congestion, and it's possible for high bandwidth applications to trigger congestion collapse, unless they implement congestion control measures at the application level.
Uncrossing Trades (UTs)	On Book trades executed during any auction uncrossing.
VWAP	Volume weighted average price.
IPv4	IPv4 is a connectionless protocol for use on packet-switched Link Layer networks (e.g., Ethernet). It operates on a best effort delivery model in that it does not guarantee delivery, nor does it assure proper sequencing or avoidance of duplicate delivery. These aspects, including data integrity, are addressed by an upper layer transport protocol (e.g., Transmission Control Protocol).
Cross Order Trade	A trade resulting from the submission of a Cross Order by market participants that results only in a trade and has no impact to orders.
Pegged Order	A hidden order pegged to the mid-point of the best bid and offer price or pegged to the best bid(offer) for instrument
Pegged Limit Order	A pegged order with a stop price also known hard limit.
EOD Volume Auction Uncrossing	A dark auction call which is triggered at end of the day after the CPX session. The uncrossing will happen at the closing price

2 OVERVIEW

~~Market data will be published through various services with each service disseminating many different types of market data.~~

~~Please read this specification document in conjunction with the JSE Market Overview document for further detail and clarification.~~

~~Each service will be available via one or more FAST market data gateways disseminating different types of market data in many different ways.~~

~~Each~~The market data feed is a stream of FAST encoded FIX messages which will provide one or more of the following real-time information for each instrument traded on the ~~System for the~~ JSE and NSX Equity Markets.

a) Security Definition
b) Security Status
c) Best Bid Price and Volume
d) Best Offer Price and Volume
e) On Book and Off Book Trades
f) Indicative Auction Information
g) Opening Price
h) Closing Price
i) Market Operations Announcements
j) No. of Trades (On Book)
k) No. of Trades (Off Book)
l) Volume of Trades (On Book)
m) Volume of Trades (Off Book)
n) Turnover (On Book)
o) Turnover (Off Book)
p) Trade High/ Low (On Book)
q) VWAP (On Book)
r) VWAP (All Trades – i.e. On Book and Off Book)

i) Price depth information for the top of the order book. The feed provides information on the aggregated displayed quantity and the number of displayed orders per price point for a book depth provided by each market data service.

ii) Price, volume, trade type, and time for each executed On Book trade that is published to the market. (Date is not published for On Book trades as it will be the current trading date which users will be aware of).

iii) Price, volume, trade type, date and time of each confirmed Off Book trade that is published to the market.

iv) Indicative Auction Information which is the auction uncrossing price and the associated trade volume.

v) Statistics of the Instrument (e.g. high/low, volume, VWAP, etc.).

vi) Trading status of the instrument.

vii) Official previous closing price, opening price and closing price.

viii) Market Operations Announcements

In addition, each feed enables participants to download the Security Definition messages for all instruments.

The feed is a multicast service based on the technology and industry standards UDP, IPv4, FAST and FIX. The application messages are defined using the FIX 5.0 (Service Pack 2) standard and comply with the best practices outlined by the FIX Market Data Working Group. The data feed is transmitted in the FAST v1.1 encoding method to minimize bandwidth and reduce latency and conforms to Level 1 of the FAST 1.1 Session Control Protocol Specification. Clients are advised to refer to the FAST 1.1 Session Control Protocol Specification when developing the interfaces that should connect to the JSE.

~~The number of gateways disseminating market data will depend on the different types of market data provided by each service and the number of instruments for which a service provides market data as the market data feed will be load balanced by instruments.~~

~~The following market data services will be available for the JSE.~~

- ~~JSE Reference Data Full (FTP)~~
- ~~Level 1 BBO service (FIX/ FAST)~~
- ~~Level 2 Full Depth (ITCH)~~
- ~~Regulatory News (FIX/ FAST)~~
- ~~Indices service (FIX/FAST)~~

~~The market data services available in the System for the NSX are as follows:~~

- ~~JSE Reference Data Full (FTP)~~
- ~~Level 1 BBO service (FIX/ FAST)~~
- ~~Level 2 Full Depth (ITCH)~~
- ~~Regulatory News (FIX/ FAST)~~
- ~~Indices service (FIX/FAST)~~

The server caters for adequate recovery and replay procedures for Market Users to recover any missed messages or out of sequence messages in the event of a failure at the JSE or within a Market User environment. These recovery and replay interfaces are point-to-point services based on the technology and industry standards TCP/IP, FIXT and FIX.

~~JSE Reference Data Full (FTP)~~

~~The reference data defined will be made available for both the JSE and NSX markets via the FTP reference data download. Please refer to Volume 09 – JSE Reference Data Management for more details of this service.~~

~~LEVEL 1 BBO SERVICE:~~

~~The service provides the order book, statistics, On Book trades, Off Book trades, list of instruments traded in the system and trading status of the instruments. This service will be available via the FAST market data feed for the JSE and NSX markets. The following will be available via the Level 1 BBO Service:~~

The order book information of this service will be disseminated real time via the Market Data Incremental Refresh message and will be limited to the top of the order book published by price.

The matrix shows the minimum number of FAST market data gateways which will be required to provide each of the market data services. This matrix also briefly includes the relevant JSE configurations for each of the Gateways.

Service	Level 1 Service (JSE)	Level 1 Service (NSX)
	(GW 1) MBP1	(GW 1) MBP1
Book Update Type	Incremental	Incremental
Book Type	MBP	MBP
Book Depth	1	1
Statistics Update Type	Incremental	Incremental
Statistics Type	All	All
Trades	All Published Trades	All Published Trades
Recovery Channel	TCP	TCP
Replay Channel	TCP	TCP
Status	All Statuses	All Statuses

~~In addition, the following Market Data Services are available.~~

~~**LEVEL 2 FULL DEPTH SERVICE (ITCH)**~~

~~This service will be available via the ITCH market data feed for the JSE and NSX markets.~~

~~The full depth order book will be available in incremental updates to the market published by order.~~

~~This service will be available real-time via the ITCH market data feed for the JSE and NSX markets. Please refer to Volume 05 - Market Data Feed (ITCH - UDP) for more details.~~

~~**REGULATORY NEWS**~~

~~News obtained from the JSE SENS system (SENS) will be disseminated to the JSE market via the JSE Regulatory News Feed.~~

~~News obtained from the NSX NENS system (NENS) will be disseminated to the NSX market via the NSX Regulatory News Feed~~

~~This service will be available via the FAST Regulatory News feed. Please Refer to Volume 08 - Regulatory News Feed (FAST - UDP) for more details.~~

~~**INDICES**~~

~~Real time index values and their statuses (including the total return value) for the JSE indices that are obtained from FTSE will be disseminated to the JSE market via the JSE Indices Feed.~~

~~Real time index values and their statuses (including the total return value) for the NSX indices that are obtained from FTSE will be disseminated to the NSX market via the NSX Indices Feed.~~

~~This service will be available via the FAST indices feed. Please Refer to Volume 07 - Indices Feed (FAST - UDP) for more details.~~

3 FAST GATEWAY SERVICE DESCRIPTION

3.1 System Architecture

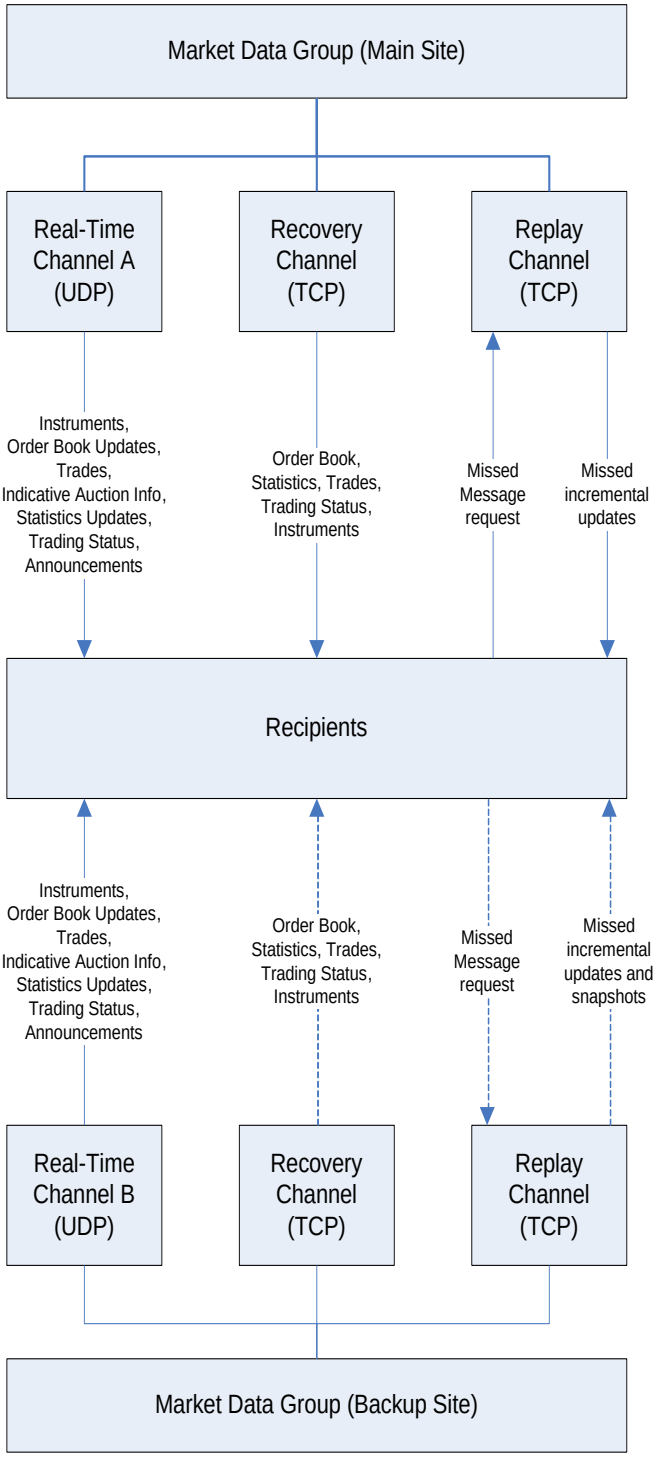
The market data feed is load balanced by market data group.

While each group will contain multiple instruments, each instrument is assigned to just one market data group. While the group an instrument is assigned to may change from day to day, it will not change within a day.

Each market data group includes a multicast Real-Time channel for the dissemination of real-time market data.

Two TCP recovery channels are available per market data group; Recovery and Replay.

While a recipient may connect to the Replay channel to recover from a small data loss, it should use the Recovery channel after a large data loss (i.e. late joiner, major outage, etc.).



3.1.1 Real-Time Channel

The Real-Time Channel is the primary means of disseminating market data. Real-time updates to instruments and all market data supported by the feed are available on this multicast channel.

The list of active instruments in the market data group will be broadcast at the start of the trading day via the Security Definition message. Real-time updates of the trading status of instruments will be disseminated via the Security Status message.

Real-time updates to order books will be published via the Market Data Incremental Refresh or the Market Data Snapshot (Full Refresh) message depending on the service which provides the data. The message used in the Real-time channel will depend on the order book and the statistics update type of the market data service. If the update type is Incremental, the data will be published via Market Data Incremental Refresh message while the Market Data Snapshot (Full Refresh) will disseminate data if the update type is snapshot.

Real-time updates disseminated in snapshots will be disseminated every SNAPSHOT_PUBLISH_INTERVAL <500> milliseconds given changes have occurred in the relevant data. If no changes have occurred in the top of the order book or statistics within

the SNAPSHOT_PUBLISH_INTERVAL <500> milliseconds period, no Market Data Snapshot (Full Refresh) message will be disseminated during that period.

Real time updates to trades, statistics and indicative auction information will be published via the Market Data Incremental Refresh message.

While each Market Data Incremental Refresh and Market Data Snapshot (Full Refresh) includes a channel specific message sequence number in the field ApplSeqNum (1181), each market data entry in the message includes an instrument specific sequence number in the field RptSeq (83). The channel and instrument level sequence numbers are to reset to 1 at the start of each day.

The Real-time channel will also disseminate Market Operations announcements via News messages.

The server will use the Heartbeat message to exercise the communication line during periods of inactivity. A Heartbeat will be sent every HEART_BEAT <2> seconds when the Real-Time channel is inactive.

Recipients have access to two identically sequenced Real-Time feeds; one from the main site (Feed A) and one from the backup feed (Feed B). It is recommended that recipients process both feeds and arbitrate between them to minimise the probability of a data loss. The sequence numbers of both feeds will be identical. This will ensure that Recipients connecting to the backup feed will not have varying sequence numbers if they connect to the replay channel of the backup feed.

3.1.2 Recovery Channel

The TCP Recovery channel should be used by Recipients to recover from a large-scale data loss (i.e. later joiner or major outage).

The TCP Recovery channel permits recipients to request a snapshot of the order book and statistics for any active instrument in the market data group as well as its current trading status. In addition, it enables recipients to request the retransmission of the trades published during the last MAX_TRADE_HISTORY_DURATION <60> minutes on the Real-Time channel. It also enables recipients to download the list of active instruments in the market data group.

All messages sent by the server are transfer encoded in terms of the FAST protocol. While all application messages sent by the server (e.g. Market Data Snapshot (Full Refresh)) are field encoded, the administrative messages it sends (e.g. Logon, Heartbeat, etc.) are not. All messages (i.e. both administrative and application) initiated by the client should be transfer encoded but not field encoded.

While a Recovery channel is available from the backup feed, it will only be activated in the unlikely event of an outage at the main site.

3.1.3 Replay Channel

The TCP Replay channel should be used by Recipients to recover from a small-scale data loss.

The TCP Replay channel permits recipients to request the retransmission of REPLAY_CACHE_SIZE <250,000> messages already published on the Real-Time (multicast) channel.

The Replay channel supports the retransmission of the last REPLAY_CACHE_SIZE <250,000> messages published on the Real-Time channel. The channel does not support the retransmission of messages published on the Recovery channel or from previous trading days.

All messages sent by the server are transfer encoded in terms of the FAST protocol. While all application messages sent by the server (e.g. Market Data Incremental Refresh, Security Definition, etc.) are field encoded, the administrative messages it sends (e.g. Logon, Heartbeat, etc.) are not. All messages (i.e. both administrative and application) initiated by the client should be transfer encoded but not field encoded.

While a Replay channel is available from the backup feed, it will only be activated in the unlikely event of an outage at the main site.

3.2 Message Overview

The market data feed utilises the FIX application messages described below to disseminate instruments, market data, and Market Operations announcements for the JSE and NSX markets.

Message	Description	Usage (By Channel)		
		Real-Time	Recovery	Replay
Security Definition	Used to disseminate details (Security ID (Instrument ID), Symbol, ISIN, TIDM, Segment, Instrument Status, Order Books Corporate Action Indicators (Ex-Markers and Annotations)) on all active and suspended instruments in the System each day. One message is sent per instrument and each message will only contain the details of one instrument.	√	√	√
Security Status	Used to communicate the trading status as the instrument moves from one trading session to another (e.g. Pre-Open, Regular Trading, Halt, etc.) of all active and suspended instruments per order book. There will be two order books for each instrument i.e. Normal Order Book (automated trading) and Off Order Book (trade reporting). This is also used to indicate a shortening or extension of any trading sessions.	√	×[1]	√

[1] Although trading status message is not disseminated in the Recovery channel the snapshot messages disseminated in the channel will indicate the security's trading status

Market Data Incremental Refresh	Used to provide: (i) An update to the order book.[2] (ii) Information on an executed trade. (iii) The indicative auction information (Indicative auction price and Indicative auction volume) (iv) Statistics (e.g. closing price, low, high, volume, VWAP, etc.). While a single message may contain multiple market data entries, it will only contain information for a single instrument per market.	√	√	√
Market Data Snapshot (Full Refresh)	Used to disseminate an order depth or price limited snapshot of the order book and statistics (e.g. low, high. Volume, VWAP, etc.) for an instrument per market.	√	√	√
News	Used to publish JSE Market Operations announcements.	√	×	√

[2] In the level 1 service the top of the order book updates are sent via Market Data Incremental Refresh messages.

3.3 Overview of a Trading Day

3.3.1 Trading on the Order Book

The regular day for On Book trading will, consist of many trading sessions: Start Of Trading, Opening Auction Call session, Continuous Trading, Halt, Halt and Close, Pause, Closing Price Publication, Closing Price Cross, Post-Close, etc. The start time for each of these sessions may vary from one set of instruments to another. A Security Status message will be published on the Real-Time channel to indicate when a particular session has commenced for an instrument. The Security Status messages that will be published for On Book trading will have MDSubBookType (1173) of On Book (1).

Users on the FAST gateway will be aware that the system is ready for the start of trading as it is sent out via a SecurityTradingStatus (326) of Start of Trading (100) session via the Security Status message. This session is prior to the Opening Auction Call. Therefore Users will be aware that the system is started and ready for trading once the Start of Trading session is disseminated. Users on the FAST gateway will be aware that the system is nearing the end of day as it is sent out via a SecurityTradingStatus (326) of Market Close (18) session via the Security Status message.

The order book of each instrument will be published by order or by price determined by Book type parameter. The JSE and NSX Markets will publish order book information by price.

The order book of each instrument will be published with Limit orders and Market orders.

The System will publish one order book for orders per instrument.

The following information will be published real-time for each entry (i.e. - price point or order, depending on the Book Type) in the order book:

(a) Unique Identifier of the instrument

(b) Price (e.g. Price of unit)

(c) Visible Size

(d) Side of the order (i.e. buy, sell)

(e) Order Code which is the Order ID of an order as assigned by the System

(f) Time indication.

3.3.2 Trade Reporting

The JSE and NSX ~~allows~~Equity Markets allow trade reporting (reporting of Off Book trades) from *8:00* to *18:15* each trading day.

As per the JSE and NSX Equity Market rules, Off Book trades can be reported for suspended instruments. Hence, Off Book statistics will be published for such suspended instruments even after publishing its closing price upon suspension.

At the start of the reporting period a Security Status message, with a SecurityTradingStatus (326) of Start Trade Reporting (17) and an MDSubBookType (1173) of Off Book (2), will be broadcast for each instrument for which trade reporting is permitted.

Similarly, at the end of the reporting period a Security Status message, with a SecurityTradingStatus (326) of End Trade Reporting (130) and an MDSubBookType (1173) of Off Book (2), will be broadcast for each instrument.

A Security Status message, with a SecurityTradingStatus (326) of Market Close (18) and an MDSubBookType (1173) of Off Book (2), will be broadcast for each such instrument at the Market Close.

3.3.3 List of Instruments

A Security Definition message will be broadcast for each active and suspended instrument on the Real-Time channel as follows using the parameter SEC_DEF_DELAY_FROM_SOD:

Level 1 BBO service – JSE	4 minutes after the Start of Day
Level 1 BBO service – NSX	5 minutes after the Start of Day

There will be no Security Definition message published for inactive instruments.

3.3.4 Ex-Marker and Annotation Information

The Ex-Marker and Annotation information will be broadcast real time through the Security Definition message via the CorporateAction (292) field.

If Ex-Marker and Annotation information associated with an instrument is updated during the inter-day period, the notification to the market will be published at market start of the next trading day.

If Ex-Marker and Annotation information associated with an instrument is updated during the intra-day period, the notification to the market will be published real time at the time of associating the Corporate Action Indicators with the instrument.

The Ex-Marker and Annotation information with regard to a particular instrument will be disseminated in a particular format as described below.

<Ex-MarkerID(XX)><EffectiveFromDate(YEARMMDD)><EffectiveToDate(YEARMMDD)>

As a single instrument may contain multiple Ex-Markers and/or Annotations, the CorporateAction (292) field will publish the information in a string format delimited by a space as per the example below.

GT2011041920110520 XD2011010520110705

3.3.5 Trading Status

The Security Status message will disseminate the trading status changes of each instrument real time as and when the instrument moves from one session to another during the trading day. This is also used for any shortening or extension of any trading sessions including instruments halts, market order imbalances and price monitoring extensions.

3.3.6 Trading Halt

Trading in an instrument may be halted during the day by JSE Market Operations.

The Security Status message will be published to indicate when a particular instrument is halted manually. The Security Status message will be published with Halt (2) as the SecurityTradingStatus (326). The reason for the halt and that it applies to On Book trading will be specified in the HaltReason (327) and MDSubBookType (1173) fields respectively.

When automated trading is resumed a Security Status message will be published with the appropriate status in the SecurityTradingStatus (326) field (i.e. Re-Opening Auction Call (105), Regular Trading (17), etc.) and MDSubBookType (1173) of On Book (1) for On Book trading.

Trade reporting in an instrument can also be halted. The Security Status message will be published to indicate when trade reporting for a particular instrument is halted manually. The Security Status message will be published with Halt (2) as the SecurityTradingStatus (326). The reason for the halt and that it applies to Off Book trading will be specified in the HaltReason (327) and MDSubBookType (1173) fields respectively.

When trade reporting is resumed, a Security Status message will be published with the current trading status and MDSubBookType (1173) of Off Book (2).

A trading halt will not be carried forward to the next trading day.

3.3.7 Pause

JSE Market Operations may also manually trigger the pause session during the day. Trading of an instrument being paused can be resumed to the current trading status.

The Security Status message will be published to indicate when a particular instrument is paused. The Security Status message will be published with Pause (111) as the SecurityTradingStatus (326).

When trading is resumed a Security Status message will be published with the current trading status in the SecurityTradingStatus (326) field.

3.3.8 Trading Suspension

An instrument may be suspended during or outside trading hours. The suspension may be lifted later in the day or it may be carried forward to subsequent trading days. A Security Definition message with a SecurityStatus (965) of Suspended (9) will be published if an instrument is suspended. A suspension only applies to On Book trading.

If at the start of a trading day, an instrument is still in a suspended state it will still be included in the Security Definition message published by the server.

A Security Definition message with a SecurityStatus (965) of Active (1) will be published if the suspension is lifted during trading hours. Separate Security Status messages will also be published if On Book trading and/or Off Book trade reporting is enabled for the instrument and the value in the MDSubBookType (1173) field will indicate that it is On Book.

3.3.9 Intra-Day Trading Session Updates

3.3.9.1 Extension of an Auction Call Session

An auction call session (Volatility, Intra-Day, etc.) may be extended due to a market order imbalance or if the current auction price is significantly different from the last sale (price monitoring extension). An auction call session could also be extended or even shortened manually by JSE Market Operations.

Upon such events, a Security Status message will be broadcast with the value Market Order Imbalance (1) or Price Outside Range (100) or Extended by Market Operations (101) (in case of a manual extension) or Shortened by Market Operations (102) (in case of manual shortening) in the field SecurityTradingEvent (1174). This message will also indicate whether the change applies to On Book trading or Off Book trade reporting in the MDSubBookType (1173) field and the new time at which the auction will take place in the Text (58) field and the current security status in SecurityTradingStatus (326).

3.3.9.2 Adjustment of other trading sessions by Market Operations

JSE Market Operations may manually extend or shorten a particular trading session. In such a case, a Security Status message will be broadcast with the value Extended by Market Operations (101) or Shortened by Market Operations (102) in the field SecurityTradingEvent (1174). This message will also indicate whether the change applies to On Book trading or Off Book trade reporting in the MDSubBookType (1173) field and the new time at which the session will end in the Text (58) field and the current security status in SecurityTradingStatus (326).

3.3.9.3 Automatic trading session updates

A trading session (Regular Trading, Opening Auction Call, etc.) may be changed automatically to another session (Volatility Auction Call, Halt) due to a circuit breaker breach

caused by volatile trading. In such a case, a Security Status message will be broadcast with the value Circuit Breaker Tripped (103) in the field SecurityTradingEvent (1174). The message will indicate whether the change applies to On Book trading or Off Book trade reporting MDSubBookType (1173) field and this message will also include the new time at which the session will end in the Text (58) field and the current security status in SecurityTradingStatus (326).

3.4 Order Book Management (~~Price Depth~~Level 1 Service)

The market data feed provides recipients with a view of the level 1 of the order book, where all limit orders at the first price point are aggregated~~ at each price level, for a configurable number of price points. The feed provides the aggregate displayed quantity and the number of represented limit orders at each price level.~~. This information is broadcast as incremental updates~~ (disseminated only in the level 1 service)~~ or as a snapshot disseminated at a configurable time period if there is an update on the Real-Time channel. ~~The configurable number of price points for the JSE is still to be determined.~~

3.4.1 Incremental Refresh

A Market Data Incremental Refresh message will be published to update the order book in the level 1 service where only the top of the order book is published. For the service levels, a message may contain multiple market data entries each of which could add, change or delete one side of ~~one~~the top price level. Each entry includes the field MDPriceLevel (1023) to indicate the price level being updated and this will be set to 1 for the level 1 service. A single message may include entries for multiple instruments. A client can identify multiple entries in a message by looking at the field encoding of the FAST message.

3.4.1.1 Adding a Price Level

When ~~a new~~the top price level is created in an order book on any side, a Market Data Incremental Refresh message will be broadcast with an MDUpdateAction (279) of New (0). The message will include the price, aggregate displayed quantity and number of orders for the top price level.

The field MDPriceLevel (1023) will indicate the display position of the price level being added~~. All rows in~~ and this will be set to 1 for the ~~order book below the new price~~ level ~~should be pushed down. The recipient's application should automatically re-number the price levels below the newly added price level. For example, if there were already 20 price levels for a~~1 service~~ configured to display only 20 price points, recipients should also delete the 21st price level from their applications.~~

~~If MDPriceLevel (1023) is set to "1" the message will update the top of the order book.~~. The recipient's application should ensure that there are no prices higher than ~~this~~the top price level.

Market Data Incremental Refresh message will not be broadcasted for hidden pegged or pegged limit orders.

3.4.1.2 Changing a Price Level

When ~~an existing~~the top price level is changed a Market Data Incremental Refresh message will be broadcast with an MDUpdateAction (279) of Change (1). The field MDPriceLevel (1023) will ~~indicate which price level is being updated. The message will include the price, quantity and number of orders~~be set to 1 for the ~~updated price~~ level 1 service. The recipient's application should ensure that the top level of the ~~equivalent price level~~order book is updated to reflect the values provided in the message.

3.4.1.3 Deleting a Price Level

When the existing top price level is removed a Market Data Incremental Refresh message will be broadcast with an MDUpdateAction (279) of Delete (2). The field MDPriceLevel (1023) will indicate ~~which~~that the top price level is being removed.

~~All rows in the order book below the deleted price level should be pushed up. The recipient's application should automatically re-number the price levels that were below the deleted price level.~~ The server will separately publish an update to add a new price level at the ~~bottom~~top of the order book.

A '-1' will be received in the field MDEntryPX (TAG 270) on the Market Data Incremental Refresh when orders that determined a Closing Price are removed during End Market. This should be disregarded as it is not an update to the closing bid and offer values.

3.4.2 Snapshot

Market Data Snapshot (Full Refresh) messages will be used to disseminate a limited view of each order book. The snapshot will include the price, aggregate display quantity, number of orders and display position for each price level.

A snapshot will be published even if there are no orders in the order book. In such a case, the Market Data Snapshot (Full Refresh) message will include a MDEntryType (269) of Empty Order Book (J).

MDPriceLevel (1023) will be maintained for order books per instrument.

An order book snapshot may span across multiple Market Data Snapshot (Full Refresh) messages. In such cases, the LastFragment (893) field will be “N” for all but the final message for the instrument. The final message will include a LastFragment (893) of “Y”.

~~While the~~The vast majority of snapshot messages are disseminated via the Snapshot channel. However, in certain scenarios (e.g. recovery after a JSE server failure), they may also be included in the Real-Time channel. . Snapshot messages may be included in the real time feed when the JSE server goes through a failover. The snapshot ensures that recipients are aware of the current snapshot of the market. Snapshots on the Real-Time channel should be processed by recipients.

3.4.3 Market Orders

The aggregate displayed quantity of market orders, if any, on each side of the order book, along with the number of represented orders, is disseminated during each Auction Call session.

In the case of market orders, the relevant market data entry of a Market Data Incremental Refresh or Market Data Snapshot (Full Refresh) message will include a MDEntryType (269) of Market Bid (b) or Market Offer (c). Such entries will not include a MDPriceLevel (1023) or a MDEntryPx (270).

3.5 Time and Sales

The market data feed provides recipients with the price, volume and time for each executed trade. Details of trade cancellations and corrections will also be broadcast.

When a trade is executed a Market Data Incremental Refresh message will be broadcast with a MDUpdateAction (279) of New (0) and MDEntryType (269) of Trade (2). The trade price, quantity and time will be specified in the fields MDEntryPx (270), MDEntrySize (271) and MDEntryTime (273) respectively.

Each trade will include a unique Trade ID in the field MDEntryID (278) which will be referenced if the trade is cancelled or corrected. The recipients will be able to identify their own trades with this Trade ID while ensuring anonymity as the same Trade ID will also be tagged in the trade capture reports carrying the relevant trade information in the Post Trade gateway.

For automated trades (ATs) and Off Book trades the MatchType (574) is not published. An uncrossing trade (UT) or EOD Volume Auction Trade (VT) is identified via the TradeCondition (277) and MatchType (574) fields. An Off-Book Trade or Cross Order is identified via the TrdSubType (829) field.

The Market Data Incremental Refresh message will be broadcasted with an MDUpdateAction (279) of New (0) and MDEntryType (269) of Trade (2) whenever a hidden Pegged or hidden Pegged Limit order is fully or partially filled during regular trading. Off Book Trades

In the case of an Off Book trade, the trade type will be indicated in the TrdSubType (829) field of the Market Data Incremental Refresh message. The date and time the trade was agreed between the firms will be specified in the fields MDEntryDate (272) and MDEntryTime (273) respectively.

3.5.1 Auction Trades

The Market Data Incremental Refresh message will contain a MatchType (574) of Auction (5) and a TradeCondition (277) of Opening Price (R), Volatility (w), Closing Price (AJ), Re-Opening Auction (AZ),Intra-Day Auction (7) , EOD Volume Auction Call (VU) or Futures Close out Auction (FC) if the trade was executed in an Opening, Volatility, Closing, Re-Opening Intra-Day, EOD Volume Auction Call or Futures Close out Auction respectively.

3.5.2 Trade Cancellation and Corrections

If a trade is cancelled a Market Data Incremental Refresh message will be broadcast with a MDUpdateAction (279) of Delete (2) and a MDEntryType (269) of Trade (2). The message will contain the Trade ID of the cancelled trade and the cancelled time in the fields MDEntryID (278) and MDEntryTime (273) respectively.

If a trade is corrected a Market Data Incremental Refresh message will be broadcast with a MDUpdateAction (279) of Change (1) and a MDEntryType (269) of Trade (2). The message will contain the Trade ID of the corrected trade, the corrected price, the corrected quantity and the corrected time in the fields MDEntryID (278), MDEntryPx (270), MDEntrySize (271) and MDEntryTime (273) respectively. This JSE will not be using this functionality.

A trade cancellation is final (i.e. once a trade is cancelled it will not be reinstated). A corrected trade may subsequently be corrected again or busted.

3.6 Indicative Auction Information

The market data feed provides recipients with the indicative auction price as well as the indicative executable auction volume. The update frequencies and times are configurable and may be changed. Indicative Auction Information will be published real time during the following auction call sessions:

(a) Opening Auction Call
(b) Intra-day Auction Call
(c) Volatility Auction Call
(d) Re-Opening Auction Call
(e) Closing Auction Call
(f) Futures Close Out Auction Call

During the above mentioned auction call sessions a Market Data Incremental Refresh message will be broadcast with one market data entry. This entry will be the indicative auction price and the indicative executable auction volume at this price.

The first entry will contain an MDUpdateAction (279) of New (0), an MDEntryType (269) of Auction Clearing Price (Q) and an MDQuoteType (1070) of Indicative (0). The indicative auction price and the executable quantity will be specified in the fields MDEntryPx (270) and MDEntrySize (271) respectively. The entry will not include an MDEntryPx (270) and MDEntrySize (271) if an indicative auction price does not exist (i.e. the order book is not locked or crossed).

Note: Indicative Auction Information will not be published for the EOD Volume Auction

3.7 Statistics

The JSE will publish the instrument statistics outlined in this section for the current trading day only. The stats for JSE will be independent of that of the NSX.

All On Book trade types will update all On Book market data statistics.

Statistics calculated will be rounded down to three decimal places.

Certain Off Book trades will update the following statistics:

(a) Turnover (Off Book)
(b) VWAP (All Trades i.e. On Book and Off Book)
(c) Volume (Off Book)
(d) Number of Trades (Off Book)

For High Price only On Book Trades will be used for calculating statistics (Off Book trades will not be included).

For Low Price only On Book Trades will be used for calculating statistics (Off Book trades will not be included).

The Previous Close will only be published at the start of day.

The market data feed provides recipients with the following statistics for each instrument:

(i) Opening price for the day.
(ii) Closing price for the day.
(iii) Highest traded price for the day (For On Book trades only).
(iv) Lowest traded price for the day. (For On Book trades only)
(v) Volume of Off Book trades for the day.
(vi) Volume of On Book trades for the day.
(vii) VWAP of all trades (On Book and Off Book) for the day.
(viii) VWAP of On Book trades for the day.
(ix) Turnover of the Off Book trades for the day.
(x) Turnover of the On Book trades for the day.
(xi) Number of Off Book trades for the day.
(xii) Number of On Book trades for the day.
(xiii) Closing bid price of the day.
(xiv) Closing offer price of the day.
(xv) Previous Closing price.

Trade Cancellations and Corrections performed will update the above mentioned statistics which will be reflected via the Statistics message.

3.7.1 Incremental Update

The above information will be broadcast in real-time and will be disseminated as separate market data entries via the Market Data Incremental Refresh message. The MDUpdateAction (279) of all these entries will be New (0). The MDEntryType (269) of each entry will indicate the particular statistic that is being updated.

In the event a statistic (e.g. closing price) is amended by the JSE, a Market Data Incremental Refresh message will be transmitted with the corrected value. The entry will not include an MDEntryPx (270) or MDEntrySize (271) in the unlikely event a previously published statistic is withdrawn (e.g. closing price). The only statistics that JSE Market Operations can amend are High Price, Low Price, Opening Price and Closing Price.

3.7.2 Snapshot

Snapshots of statistics will be disseminated via the Market Data Snapshot (Full Refresh) message. The MDEntryType (269) of each entry will indicate the statistic being updated.

A snapshot will be published even if there are no statistics for an instrument. In such a case, the Market Data Snapshot (Full Refresh) message will include an MDEntryType (269) of No Statistics (z).

While the vast majority of snapshot messages are disseminated via the Snapshot channel, in certain scenarios (e.g. recovery after a server failure), they may also be included in the Real-Time channel. Snapshots on the Real-Time channel should be processed by recipients.

3.8 Quotation Conventions

The values specified in the MDEntryPx (270) field of the Market Data Snapshot (Full Refresh) and Market Data Incremental Refresh messages should, depending on the quotation convention for the instrument, be interpreted as price per share.

The quotation convention for each instrument will be disseminated via the PriceType (423) field of the Security Definition message.

3.9 Market Operations Announcements

JSE Market Operations announcements are included in the market data feed. In addition to the actual text of the announcement and an associated headline or subject, recipients are provided with its urgency, the time it was generated and the list of instruments (if necessary) and underlying instruments, if any, to which ~~it~~the announcement relates.

Announcements are disseminated via the News message.

4 CONNECTIVITY

4.1 Transmission Standards

4.1.1 Multicast Channels

The Real-Time channel utilise IP version 4 (IPv4) over UDP and Ethernet standards. UDP header information will be as defined in the IETF RFC 791 (IPv4) and RFC 768 (UDP) transmission protocol standards. One or more FAST encoded FIX messages may be included in a single UDP packet.

4.1.2 Point-to-Point Channels

The Recovery and Replay channels utilise IP version 4 (IPv4) over TCP and Ethernet standards. TCP header information will be as defined in the IETF RFC 793 standard and IPv4 will be as defined in the RFC 791 standard.

4.2 Application IDs (ApplIDs)

4.2.1 Server

The ApplID will be different and defined separately for each FAST Market Data gateway primary and secondary server. The server ApplIDs for the Real-Time, Recovery and Replay channels of each market data group attached to different gateways of all market data services are given below.

Market Data Group	Real-Time Channel		Snapshot Channel		Replay Channel	
	Primary	Secondary	Primary	Secondary	Primary	Secondary
JSE Level 1	*<JSELVL1P>*	*<JSELVL1S>*	*<JSELVL1P>*	*<JSELVL1S>*	*<JSELVL1P>*	*<JSELVL1S>*
NSX Level 1	*<NSXLVL1P>*	*<NSXLVL1S>*	*<NSXLVL1P>*	*<NSXLVL1S>*	*<NSXLVL1P>*	*<NSXLVL1S>*

4.2.2 Clients

The Interface User ID (CompID) and the IP address of each client wishing to connect to the Recovery and Replay channels must be registered with JSE before communications can begin. An Interface User ID (CompID) may, at any particular time only be logged into one, Recovery channel and one Replay channel TCP channel across all market data groups.

4.2.2.1 Passwords

Each new Interface User ID (CompID) will be assigned a password on registration. Clients will be required to change the password on first use to one of their choosing via the Logon message. The acceptance of a login request indicates that the new password has been accepted. The new password will, if accepted, be effective for subsequent logins. The JSE will publish a separate document on the JSE requirements for passwords.

Depending on the JSE password policy implemented, the password of each Interface User ID (CompID) may need be changed at least every *<30>* days. If not, the password will expire and the client will be unable to login to the server. In such a case, the client should contact JSE to have its password reset. Again depending on the JSE password policy, the SessionStatus (1409) of the server's Logon message will be Password Due to Expire (2) for the last <5> days of a password's validity period.

4.3 Production IP Addresses and Ports

The FAST market data feed is load balanced by market data group. While each group will contain multiple instruments, each instrument is assigned to just one market data group. While the group an instrument is assigned to may change from day to day, it will not change within a day. The Security Definition messages available on the Real-Time channel of the various market data groups may be utilized by recipients to identify the instruments assigned to each group.

Market Data groups are a logical grouping and do not affect the Market data services. Initially, the only market data grouping that will be implemented for instruments is by the relevant market. (i.e. JSE and NSX)

The production IP addresses and ports of the Real-Time, Snapshot and Replay channels are ~~detailed in the consolidated JSE Production Market Facing Client document.~~provided upon etc same comment used in MITCH

The Market Data Group identifiers will be assigned as below. Clients are required to specify the Market Data Group identifier accordingly on performing Replay Request message via the Replay Channel based on the server the client is connected to.

Market Data Group	Replay Channel	
	Primary	Secondary
JSE	*1*	*2*
NSX	*3*	*4*

5 RECOVERY

5.1 Recipient Failures

It is recommended that recipients process both Real-Time feeds (i.e. Feed A and Feed B) to minimise the probability of a data loss.

A message loss can be detected using the ApplSeqNum (1181) included in each message on the Real-Time channel. If a gap in sequence numbers is detected, the Recipient should assume that some or all of the order books and statistics maintained on its systems are incorrect and initiate one of the recovery processes outlined below.

Each entry of a Market Data Incremental Refresh or a Market Data Snapshot (Full Refresh) message disseminated on the Real-Time channel also includes an instrument specific sequence number in the field RptSeq (83). Recipients may use this instrument level sequencing to determine the instruments for which a recovery process should be initiated.

5.1.1 Recovery Channel

The TCP Recovery channel should be used by recipients to recover from a large-scale data loss (i.e. late joiner or major outage).

The channel permits recipients to request a snapshot of the order book and statistics for the active instruments in the market data group as well as their current trading status. In addition, it enables recipients to request the retransmission of the trades published during last MAX_TRADE_HISTORY_DURATION <60> minutes on the Real-Time channel. It also enables recipients to download the list of active instruments in the market data group.

Announcements can be recovered for the last MAX_ANNOUNCEMENT_COUNT *<10,000>* messages on the Real-Time channel.

Each Interface User ID (CompID) may login to the Recovery channel of a particular market data group USER_MAX_LOGINS_FOR_SNAPSHOT_CHANNEL <500> times each day. The total number of Security Definition Request and Market Data Request messages that a client may submit on the Recovery channel of a particular market data group is also limited to USER_MAX_REQUESTS_FOR_SNAPSHOT_CHANNEL <500>.

Recipients may request the JSE to reset its login and request counts. This feature is intended to help manage an emergency situation and should not be relied upon as a normal practice.

While a client may submit multiple Market Data Request, the client cannot send more than the maximum concurrent unprocessed requests at any point defined by the parameter USER_MAX_CONCURR_REQUESTS_FOR_SNAPSHOT_CHANNEL*<10>,* if a client submits multiple requests on the Recovery channel, they will be processed serially (i.e. one at a time). Active requests of multiple clients will be served on a FIFO basis.

All messages sent by the server are transfer encoded in terms of the FAST protocol. While all application messages sent by the server (e.g. Market Data Snapshot (Full Refresh), Security Definition, etc.) are field encoded, the administrative messages it sends (e.g. Logon, Heartbeat, etc.) are not. All messages (i.e. both administrative and application) initiated by the client should be transfer encoded but not field encoded.

5.1.1.1 Establishing a Connection

The client should use the relevant IP address and port to establish a TCP/IP session with the Recovery channel. The client should initiate a FIX connection by sending the Logon message. The client should identify itself by specifying its Interface User ID (CompID) in the Username (553) field. The server will validate the Interface User ID (CompID), password and IP address of the client.

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 a logon attempt fails because of an invalid Interface User ID (CompID), invalid password or IP address, or if a message is sent prior to the login being established the server will break the TCP/IP connection with the client without sending a Logout message.

The client must wait for the server's Logon before sending additional messages. Messages received from the client before the exchange of Logon messages will be ignored.

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 and then break the TCP/IP connection with the client.

The Recovery channel supports a maximum number of <150> concurrent logins across all clients. In addition, each Interface User ID (CompID) may login to the Recovery channel of a particular market data group <500> times each day. Once this limit is reached, the server will reject any new logon attempt with a Logout and then break the TCP/IP connection with the client. The SessionStatus (1409) of such a Logout message will be Logons Not Allowed (7).

If a Logon message is not received within a defined period of the establishment of a TCP/IP connection, the server will break the TCP/IP connection with the client.

If a Security Definition Request or Market Data Request is not received within USER_MAX_IDLING_TIME <5> seconds of a successful logon, the server will send a Logout message and then break the TCP/IP connection with the client. The Text (58) field of Logout will contain the value "c" (i.e. Logout Due to Inactivity). Each time the TCP/IP connection is terminated, it will increment the counter of the maximum amount of times each Interface User ID (CompID) may login to the Recovery channel.

A second attempt to log in ~~by an already logged in client will be rejected via a Business Message Reject.~~to the same Market Data Snapshot Channel or a Market Data Snapshot Channel of a different Market Data Group, by an already logged in client will be rejected immediately by breaking the connection. No message is sent to the client in this case, as the client is not authenticated. The original session is not affected by this disconnection.

5.1.1.2 Heartbeats

The server will not send heartbeats on the Recovery channel during periods of inactivity.

5.1.1.3 Requesting Instrument

Once connected to the Recovery channel, clients may use the Security Definition Request message to request the details of all active and suspended instruments in the market data group or those in the group from a particular segment. The Security Definition Request should include a unique SecurityReqID (320) and a SecurityRequestType (321) (i.e. all instruments or all instruments for a segment).

If the request is successful, the server will disseminate the details of each instrument via the Security Definition message. Each such message will include a SecurityResponseType (323) of List of Securities Returned (4) and the SecurityReqID (320) of the request. The last message sent in response to the request will include a LastRptRequested (912) of Last Message (Y). The client will not receive the details of any subsequent master file updates.

An invalid Security Definition Request will generally be rejected via a Security Definition message with a SecurityResponseType (323) of Cannot Match Selection Criteria (6). However, if the number of requests for the day has already reached or a conditionally required field is missing, a Security Definition Request will be rejected via a Business Message Reject.

5.1.1.4 Requesting Snapshot: Order Book and Statistics

Once connected to the Recovery channel, clients may use the Market Data Request message to request a snapshot of the current order book or statistics for one or more instruments in the market data group. Clients may also request snapshots for the instruments in the market data group that are assigned to one or more segments. The

Market Data Request should include an MDReqID (262) and at least one SecurityID (48) or ProductComplex (1227).

A request for an order book snapshot should include a MDEntryType (269) of Bid (0) and/or Offer (1). An order book snapshot will always contain the details both sides the order book even if the Market Data Request contained just an MDEntryType (269) of Bid (0) or Offer (1). Clients are unable to request just one side (i.e. bid or offer) of an order book.

Similarly, a request for a statistics snapshot should include one or more of the relevant entries for MDEntryType (269) (e.g. High Price (7), Low Price (8), Volume (B), etc.). A statistics snapshot will contain all available statistics even if the Market Data Request contained just some of the statistics-related entries for MDEntryType (269). Clients are unable to request for a snapshot of only a selected set of statistics (e.g. high price and low price only).

A client may request for all order book and statistics snapshot in a single Market Data Request. A single message may also be used to request snapshots for multiple instruments and/or multiple segments.

If the Market Data Request is successful, the server will disseminate a snapshot of the current order book or statistics for each instrument via a Market Data Snapshot (Full Refresh) message.

A successful Market Data Request for order book will result in disseminating snapshots of all the books maintained for the relevant instruments.

Each such message will include the MDReqID (262) of the applicable Market Data Request. The order book and statistics snapshot for a particular instrument will be disseminated via two separate Market Data Snapshot (Full Refresh) messages.

The order book snapshots of order books of a single instrument will also be disseminated via separate Market Data Snapshot (Full Refresh) messages.

Each Market Data Snapshot (Full Refresh) will reflect the order book or statistics for the instrument as per the last update on the Real-Time channel. The ApplSeqNum (1181) and RptSeq (83) of the applicable Market Data Incremental Refresh or the Market Data Snapshot (Full Refresh) disseminated in the Real-Time channel will be included in the LastMsgSeqNumProcessed (369) and RptSeq (83) fields of each Market Data Snapshot (Full Refresh) respectively.

An order book or statistics snapshot will be published even if there are no orders or statistics for an instrument. In such a case, the Market Data Snapshot (Full Refresh) message will include an MDEntryType (269) of Empty Order Book (J) or No Statistics (z).

Each Market Data Snapshot (Full Refresh) message, whether it is used to provide a snapshot of the order book or statistics in the Recovery channel, will include the current status of On Book and Off Book trading for the instrument in the field MDSecurityTradingStatus (1682).

The last message sent in response to the request will include a LastRptRequested (912) of Last Message (Y). The client will not receive any subsequent order book or statistics updates for the applicable instruments.

An invalid Market Data Request will generally be rejected via a Market Data Request Reject message. However, if the number of requests for the day has already reached the defined limit or a conditionally required field is missing, a Market Data Request will be rejected via a Business Message Reject.

5.1.1.5 Requesting Missed Trades

Once connected to the Recovery channel, clients may use the Market Data Request message to request missed trades for one or more instruments in the market data group. Clients may also request missed trades for the instruments in the market data group that are assigned to one or more segments. The Market Data Request should include a MDReqID (262) and at least one SecurityID(48) or ProductComplex (1227).

A Market Data Request for missed trades should include a MDEntryType (269) of Trade (2). The request should also include the time of the last trade on the Real-Time channel processed by the client in the MDEntryTime (273) field.

A client may request missed trades and order book and/or statistics snapshots in a single Market Data Request. A single message may also be used to request missed trades for multiple instruments and/or multiple segments.

If the Market Data Request is successful, the server will retransmit all the trades for the specified instruments and/or segments after the specified time. Each missed trade will be published via a Market Data Incremental Refresh message. Each such message will include the MDReqID (262) of the applicable Market Data Request. The corresponding message and instrument sequence numbers from the Real-Time channel will be included in the ApplSeqNum (1181) and RptSeq (83) fields of each retransmitted trade. The last message sent in response to the request will include a LastRptRequested (912) of Last Message (Y).

If the request includes an MDEntryTime (273) that is prior to that of the oldest trade in the server's cache, all eligible trades in the cache will be retransmitted. Clients are unable to recover trades not in the server's cache.

An invalid Market Data Request will generally be rejected via a Market Data Request Reject message. If no trades have been missed, the Market Data Request Reject will include a MDReqRejReason (281) of "Z" and the value "102" (i.e. requested market data unavailable, no missed messages) in the Text (58) field.

A Market Data Request will be rejected via a Business Message Reject if a conditionally required field is missing or if the number of requests limit is breached.

5.1.1.6 Requesting Missed Market Operations Announcements

A Client may request for missed News messages via the Recovery channel using a Market Data Request with MDEntryType (269) set to News (w) and the MDEntryTime (273) with the time of the last Client received News message. The NoRelatedSymbols(146) block will not be required if the MDEntryType(269) is News (w); if specified, the values will be ignored by the gateway.

A Client may optionally request missed announcements, trades, order book and/or statistics snapshots in a single Market Data Request.

Recovery of News messages will not be limited by Segment or by Instrument. Upon submitting a request for News messages to the Gateway, the Client will receive all the News messages after the specified time in MDEntryTime (273) for all symbols.

The last message sent in response to the request will include a LastRptRequested (912) of Last Message (Y). The ApplSeqNum (1181) of the News messages retransmitted to honour a Market Data Request will be set to zero (0).

If none of the announcements have been missed as per the Market Data Request submitted by the Client, the Market Data Request Reject will include an MDReqRejReason (281) of "Z" and the value '102' (requested market data unavailable) in the Text (58) field.

The Clients will be able to recover MAX_ANNOUNCEMENT_COUNT <10,000> messages via the message cache size of each Gateway.

5.1.1.7 Cancelling a Request

A client may cancel an outstanding instrument or market data request via the Security Definition Request and Market Data Request messages respectively. Such a message should include a SubscriptionRequestType (263) of Unsubscribe (2) and the SecurityReqID (320) or MDReqID (262) of the request to be cancelled. While the server will not confirm a successful cancellation, it will transmit a Business Message Reject if the request is rejected. A cancellation request submitted by a client will take priority over all the requests of the client being queued.

5.1.1.8 Termination of the Connection

If the client does not terminate the connection within USER_MAX_IDLING_TIME <*5*> seconds of the transmission of the last application message, the server will send a Logout message and then break the TCP/IP connection with the client. The Text (58) field of the Logout will contain 'Log out due to inactivity'.

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 the defined limit for the day.

5.1.2 Replay Channel

The TCP Replay channel should be used by recipients to recover from a small-scale data loss. It permits recipients to request the retransmission of a limited number of messages already published on the Real-Time channel. The channel supports the retransmission of the last REPLAY_CACHE_SIZE <250,000> messages published on the Real-Time channel.

Each Interface User ID (CompID) may login to the Replay channel of a particular market data group up to USER_MAX_LOGINS_FOR_REPLAY_CHANNEL <1000> times each day. The total number of Application Message Requests that a client may send on the Replay channel of a particular market data group is also limited to USER_MAX_REQUESTS_FOR_REPLAY_CHANNEL <1000> each day.

Recipients may request the JSE to reset its login and request counts. This feature is intended to help manage an emergency situation and should not be relied upon as a normal practice.

While a client may submit multiple Application Message Request, the client cannot send more than the maximum concurrent unprocessed requests at any point defined by the parameter USER_MAX_CONCURR_REQUESTS_FOR_SNAPSHOT_CHANNEL*<10>*, if a client submits multiple requests on the Replay channel, they will be processed serially (i.e. one at a time). Active requests of multiple clients will be served on a FIFO basis.

All messages sent by the server are transfer encoded in terms of the FAST protocol. While all application messages sent by the server (e.g. Market Data Incremental Refresh, Security Definition, etc.) are field encoded, the administrative messages it sends (e.g. Logon, Heartbeat, etc.) are not. All messages (i.e. both administrative and application) initiated by the client should be transfer encoded but not field encoded.

5.1.2.1 Establishing a Connection

The client should use the relevant IP address and port to establish a TCP/IP session with the Replay channel. The client should initiate session by sending the Logon message. The client should identify itself by specifying its Interface User ID (CompID) in the Username (553) field. The server will validate the Interface User ID (CompID), password and IP address of the client.

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

The client must wait for the server's Logon before sending additional messages. Messages received from the client before the exchange of Logon messages will be ignored.

If a logon attempt fails because of an invalid Interface User ID (CompID), invalid password or IP address, or if a message is sent prior to the login being established the server will break the TCP/IP connection with the client without sending a Logout message.

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.

If a Logon message is not received within a defined period of the establishment of a TCP/IP connection, the server will break the TCP/IP connection with the client.

The Replay channel supports a maximum number of MAX_CONCURRENT_LOGINS <150> concurrent logins across all clients. In addition, each Interface User ID (CompID) may login to the Replay channel of a particular market data group limited to 1000 times each day. Once this limit is reached, the server will reject any additional logon attempt with a Logout and then break the TCP/IP connection with the client. The SessionStatus (1409) of such a Logout message will be Logons Not Allowed (7).

If an Application Message Request is not received within USER_MAX_IDLING_TIME <*5*> seconds of a successful logon, the server will send a Logout message and then break the TCP/IP connection with the client. The Text (58) field of Logout will contain the value "c" (i.e. Logout Due to Inactivity).

Each time the TCP/IP connection is terminated, it will increment the counter of the maximum amount of times each Interface User ID (CompID) may login to the Replay channel.

A second attempt to log in to the same Market Data Replay Channel or a Market Data Replay Channel of a different Market Data Group, by an already logged in client will be rejected ~~via a Business Message Reject~~immediately by breaking the connection. No message is sent to the client in this case, as the client is not authenticated. The original session is not affected by this disconnection.

5.1.2.2 Heartbeats

The server will not send heartbeats on the Replay channel during periods of inactivity.

5.1.2.3 Requesting Missed Messages

The client is expected to transmit an Application Message Request within USER_MAX_IDLING_TIME <*5*> seconds of establishing the FIX connection.

The message should include the server ApplID of the Real-Time channel to which the retransmission request applies along with the list of messages to be resent. The ApplBegSeqNum (1182) and ApplEndSeqNum (1183) fields should be used to specify the ApplSeqNum (1181) of the first and last message in the range to be resent.

The Application Message Request can be used in four modes:

(i) To request a single message. The ApplBegSeqNum (1182) and ApplEndSeqNum (1183) should both be the message sequence number of the missed message.

(ii) To request a specific range of messages. The ApplBegSeqNum (1182) should be the message sequence number of the first message of the range and the ApplEndSeqNum (1183) should be that of the last message of the range.

(iii) To request all messages after a particular message. The ApplBegSeqNum (1182) should be the message sequence number immediately after that of the last processed message and the ApplEndSeqNum (1183) should be zero (0).

(iv) To request all messages available. The ApplBegSeqNum (1182) should be one (1) and the ApplEndSeqNum (1183) should be zero (0).

The retransmission request will be serviced from the server's cache of the last REPLAY_CACHE_SIZE <250,000> messages published on the Real-Time channel. If the retransmission request includes one or more messages that are not in the server's cache, the entire request will be rejected and no messages will be retransmitted.

5.1.2.4 Response to a Retransmission Request

The server will respond to the Application Message Request with an Application Message Request Ack to indicate whether the retransmission request is successful or not. If the request is unsuccessful, the reason will be specified in the field ApplResponseType (1348).

The total number of Application Message Requests that a client may send on the Replay channel of a particular market data group is limited to USER_MAX_REQUESTS_FOR_REPLAY_CHANNEL <1000> each day. Once this limit is reached, the server will reject any additional request via a Business Message Reject.

In the case of a successful retransmission request, the server will transmit the requested messages immediately after the Application Message Request Ack. The message sequence number and where relevant, the instrument sequence number from the Real-Time channel will be included in the ApplSeqNum (1181) and RptSeq (83) fields of each retransmitted message. Once the last of these messages is sent, the server will indicate that the retransmission is complete via an Application Message Report.

5.1.2.5 Cancelling of a Retransmission Request

A client may cancel an outstanding request via the Application Message Request message. Such a message should include an ApplReqType (1347) of Cancel Retransmission (5) and the ApplReqID (1346) of the request to be cancelled. A cancellation request submitted by a client will take priority over all the requests of the client being queued.

Application message can only be cancelled before it is processed by the system. As long as the user has not received the Application Message Request Ack he will be able to cancel the request.

5.1.2.6 Termination of the Connection

If the client does not terminate the connection within USER_MAX_IDLING_TIME <*5*> seconds of the transmission of the last missed message, the server will send a Logout message and then break the TCP/IP connection with the client. The Text (58) field of the Logout will contain 'Log out due to inactivity'.

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 the defined limit for the day.

5.2 Failures at the JSE

5.2.1 Snapshots on the Real-Time Channel

In the unlikely event of an outage of a system at the JSE, recipients may be required to refresh their order book and statistics displays for one or more instruments.

5.2.1.1 Refresh Order Book

In such a scenario the server will, on the Real-Time channel, broadcast Market Data Snapshot (Full Refresh) messages for each affected instrument per each book: with an MDEntryType (269) of Empty Order Book (J) . In such an event, recipients must discard the contents of their order books. Each such Market Data Snapshot (Full Refresh) message will include the current status of the particular order book for the particular instrument, in the field MDSecurityTradingStatus (1682).

The server will then transmit a series of Market Data Incremental Refresh messages to disseminate the current order book for each affected instrument per each book, if the gateway is configured with the book update type set to Incremental mode. If the gateway is configured with the book update type set to Snapshot mode, the server will transmit Market Data Snapshot (Full Refresh) messages to disseminate the current order book for affected instrument per each book.

5.2.1.2 Refresh Statistics

In such a scenario the server will, on the Real-Time channel, broadcast Market Data Snapshot (Full Refresh) messages for each affected instrument per each book (On Book and Off Book): with an MDEntryType (269) of No Statistics (z). In such an event, recipients must discard the contents of their statistics. Each such Market Data Snapshot (Full Refresh) message will include the current status of the particular order book for the particular instrument, in the field MDSecurityTradingStatus (1682).

The server will then transmit a series of Market Data Incremental Refresh messages to disseminate the current statistics for each affected instrument per each book, if the gateway is configured with the book update type set to Incremental mode. If the gateway is configured with the book update type set to Snapshot mode, the server will transmit Market Data Snapshot (Full Refresh) messages to disseminate the current statistics for affected instrument per each book

5.2.2 Resetting Sequence Numbers

If the market data feed is, due to the unlikely event of an outage, restarted during a trading day, the message sequence number and instrument level sequence number of the Real-Time channel will be reset to 1.

In such a case, messages sent on the Real-Time channel prior to the resetting of sequence numbers will not be available for retransmission on the Replay channel. The trades resent on the Snapshot channel, if any, will not contain an ApplSeqNum (1181) or RptSeq (83) if there were originally disseminated prior to the resetting of sequence numbers.

6 MESSAGE FORMATS AND TEMPLATES

This section provides details on the three administrative messages and twelve application messages utilized by the market data feed.

All messages sent by the server are transfer encoded in terms of the FAST protocol. While all application messages sent by the server (e.g. Market Data Incremental Refresh, Security Definition, etc.) are field encoded, the administrative messages it sends (e.g. Logon, Heartbeat, etc.) are not. All messages (i.e. both administrative and application) initiated by the client should be transfer encoded but not field encoded.

The FIX format of each is described along with the applicable FAST template.

Some of the proposed tag values are not registered with FIX as these are user defined and the FIX protocol states that fields within this range are not required to be registered. The site states this in the section The FIX Protocol Organization > Technical Resources / Specifications.

6.1 Variations from the FIX Protocol

The market data feed conforms to the FIX protocol except as follows:

(i) The Market Data Snapshot (Full Refresh) message includes the fields MDSecurityTradingStatus (1682) and MDHaltReason (1684) which were introduced in Extension Pack 106.

(ii) The SecurityStatus (965) field of the Security Definition message includes the custom value Suspended (9).

(iii) The Market Data Request message includes the field MDEntryTime (273).

(iv) The MDSubBookType (1173) field is included in the Security Status message

(v) The MDEntryType (269) field of the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh, Market Data Request and MDRequest messages includes the custom values Market Bid (b), Market Offer (c), Turnover (d), Number of Trades (e) and Previous Close (f).

(vi) Closing Bid and Offer Prices will be indicated by the use of the combination of fields MDEntryType (269) and QuoteCondition (276). Closing Bid will be indicated with MDEntryType (269) of Bid (0) and QuoteCondition (276) of Closing (O). Closing Offer will be indicated with MDEntryType (269) of Offer (1) and QuoteCondition (276) of Closing (O). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(vii) VWAP of automated trades will be indicated by the use of the combination of fields MDEntryType (269) of VWAP (9) and MDOriginType (1024) of Book (0). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(viii) Turnover of On Book trades will be indicated by the combination of the fields MDEntryType (269) of Turnover (d) and MDOriginType (1024) of Book (0). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(ix) Turnover of Off Book trades will be indicated by the combination of the fields MDEntryType (269) of Turnover (d) and MDOriginType (1024) of Off Book (1). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(x) Volume of On Book trades will be indicated by the combination of the fields MDEntryType (269) of Volume (B) and MDOriginType (1024) of Book (0). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(xi) Volume of Off Book trades will be indicated by the combination of the fields MDEntryType (269) of Volume (B) and MDOriginType (1024) of Off Book (1). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(xii) Number of On Book trades will be indicated by the combination of the fields MDEntryType (269) of Number of Trades (e) and MDOriginType (1024) of Book (0). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(xiii) Number of Off Book trades will be indicated by the combination of the fields MDEntryType (269) of Number of Trades (e) and MDOriginType (1024) of Off Book (1). This applies to the Market Data Snapshot (Full Refresh), Market Data Incremental Refresh and Market Data Request messages.

(xiv) The MDReqRejReason (281) field of the Market Data Request Reject message includes the custom value “Other” (Z).

(xv) The SecurityTradingStatus (326) field of the Security Status message includes the custom values “Closing Auction Call” (102), “Volatile Auction Call” (104), “Re-Opening Auction Call”(105), “Pause” (111), “Futures Close out” (121), “Intra-Day Auction Call” (122), “Closing Price Cross” (120), “Closing Price Publication” (125).

(xvi) The SecurityTradingEvent (1174) field of the Security Status message includes the custom value “Price Outside Range” (100), “Session Extended” (101) and “Session Shortened” (102).

(xvii) The Text (58) field of the Security Status message is used to disseminate the time at which a Volatile or Resume auction will take place. In the event an Auction Call is extended, the updated time for the auction will also be disseminated via this field.

(xviii) The HaltReason (327) field of the Security Status message contains custom values specific to JSE.

(xix) The LastRptRequested (912) field is included in the Security Definition, News, Market Data Snapshot (Full Refresh) and Market Data Incremental Refresh messages.

(xx) The Indication of how the opening and closing prices of an instrument is computed is denoted via the custom field OpenCloseIndicator (30002) available in the Market Data Snapshot (Full Refresh) and Market Data Incremental Refresh messages. The data type of this field is String.

(xxi) Custom values added as per Section 10.1 to the field TrdSubType (829) to support the trade types supported in the system.

(xxii) The MDEntryType (269) field of the Market Data Snapshot (Full Refresh) message includes the custom value No Statistics (z).

(xxiii) The TradeCondition (277) field of the Market Data Incremental Refresh message includes the custom values “Re-Opening Auction” (AZ), “Futures Close out Auction” (FC) and “Intra Day Auction” (7).

6.2 Administrative Messages

6.2.1 Logon

6.2.1.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	A = Logon
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID	N	Identifier of the server sending the message. Required if the message is generated by the server.
108	HeartBtInt	Y	Indicates the heartbeat interval in seconds.
553	Username	N	CompID of the client. Required if the message is generated by the client.
554	Password	N	Password assigned to the CompID. Required if the message is generated by the client.
925	NewPassword	N	New password for the CompID.
1409	SessionStatus	N	Status of session Required if message is generated by server. **Value** **Meaning** 0 Session Active 2 Password Due to Expire

6.2.1.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	None	
108	HeartBtInt	Unsigned Integer	None	
553	Username	ASCII String	None	
554	Password	ASCII String	None	
925	NewPassword	ASCII String	None	
1409	SessionStatus	Unsigned Integer	None	

6.2.2 Logout

6.2.2.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	5 = Logout
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID	N	Identifier of the server sending the message required if the message is generated by the server
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 Not Allowed 8 Password Expired 100 Other
58	Text	N	Reason for the logout.

6.2.2.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	None	
1409	SessionStatus	Unsigned Integer	None	
58	Text	ASCII String	None	

6.2.3 Heartbeat

6.2.3.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	0 = Heartbeat
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID	Y	Identifier of the server sending the message.
1399	ApplNewSeqNum	Y	Will contain the next application sequence (i.e. ApplSeqNum (1181) of the next application message)

6.2.3.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	None	Please refer to Section 4.2.1
1399	ApplNewSeqNum	Unsigned Integer	None	

6.3 Application Messages (Client-Initiated)

6.3.1 Security Definition Request

6.3.1.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	c = Security Definition Request
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
320	SecurityReqID	Y	Unique identifier of the request sent by the client
263	Subscription RequestType	Y	Whether the request is being cancelled. **Value** **Meaning** 0 Snapshot 2 Unsubscribe
321	SecurityRequest Type	N	Type of request. Required unless Subscription RequestType (263) is Unsubscribe (2). **Value** **Meaning** 8 All Securities 9 All Securities for a Segment
1300	MarketSegmentID	N	Indicates the segment. Please refer to Section 7.1 for the valid segments. Required if SecurityRequest Type (321) is All Securities for a Segment (9).

6.3.1.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
Standard Header				
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
320	SecurityReqID	ASCII String	None	
263	SubscriptionRequest Type	Unsigned Integer with NULL support	None	
321	SecurityRequestType	Unsigned Integer	None	
1300	MarketSegmentID	ASCII String	None	

6.3.2 Market Data Request

6.3.2.1 FIX Message

Tag	Field Name		Req	Description
35	MsgType		Y	V = Market Data Request
52	SendingTime		Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
262	MDReqID		Y	Unique identifier of the market data request sent by the client
263	Subscription RequestType		Y	Type of subscription requested. **Value** – **Meaning** 0 – Snapshot 2 – Unsubscribe
267	NoMDEntryTypes		N	Number of market data types requested. Required if Subscription RequestType (263) is Snapshot (0).
➡	269	MDEntryType	N	Indicates the type of market data requested. Required if NoMDEntryTypes (267) is specified. **Value** – **Meaning** 0 – Bid 1 – Offer 2 – Trade 4 – Opening Price 5 – Closing Price 7 – High Price (For On Book trades only) 8 – Low Price (For On Book trades only) 9 – VWAP B – Volume d – Turnover e – Number of Trades f – Previous Close w – News
➡	273	MDEntry Time	N	MDEntryTime (273) of the last processed trade or market operations announcement. The time will be specified in UTC and in the HH:MM:SS.sss format. Required if MDEntryType (269) is Trade (2) or News (w).

➡	276	QuoteCondition	N	**Value** **Meaning** O Closing Will be used to indicate Closing Bid and Offer Price together with MDEntryType (269) of Bid (0) and Offer (1) respectively.
146	NoRelatedSym		N	Number of instruments or segments market data is requested for. Required if SubscriptionRequest Type (263) is Snapshot (0).
➡	48	SecurityID	N	Identification number for the security.
➡	22	SecurityIdSource	N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier
➡	1227	Product Complex	N	Indicates the segment. Please refer to Section 7.1 for the valid segments. Required if SecurityID (48) is not specified.
➡	1173	MDSubBookType	N	Type of market data to which the request relates. Absence of this field~~, the System will treat this~~ should be interpreted as 1 (Regular).for all order books. **Value** **Meaning** 1 Regular 2 Off-Book

6.3.2.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
262	MDReqID	ASCII String	None	
263	Subscription RequestType	Unsigned Integer	None	
267	NoMDEntryTypes	Unsigned Integer	None	
269	MDEntryType	ASCII String	None	
273	MDEntryTime	ASCII String	None	
276	QuoteCondition	ASCII String	None	
146	NoRelatedSym	Unsigned Integer	None	
48	SecurityID	ASCII String	None	
22	SecurityIDSource	ASCII String	None	
1227	ProductComplex	ASCII String	None	
1173	MDSubBookType	Unsigned Integer	None	

6.3.3 Application Message Request

6.3.3.1 FIX Message

Tag	Field Name		Req	Description
35	MsgType		Y	BW = Application Message Request
52	SendingTime		Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1346	ApplReqID		Y	Client specified unique identifier of the request.
1347	ApplReqType		Y	Type of request. **Value** **Meaning** 0 Retransmission of Messages 5 Cancel Retransmission
1351	NoApplIDs		N	If specified, the value in this field should always be "1". Required if ApplReqType (1347) is Retransmission of Messages (0).
➡	1355	RefApplID	N	AppIID of the Real-Time channel for which the retransmission is requested. Please refer to Section 4.2.1 for the list of valid ApplIDs. Required if NoApplIDs (1351) is specified.
➡	1182	ApplBeg SeqNum	N	ApplSeqNum (1181) of the first message in the range to be resent from the Real-Time channel. Required if NoApplIDs (1351) is specified.
➡	1183	ApplEnd SeqNum	N	ApplSeqNum (1181) of the last message in the range to be resent from the Real-Time channel. Required if NoApplIDs (1351) is specified.

6.3.3.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
Standard Header				
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1346	ApplReqID	ASCII String	None	
1347	ApplReqType	Unsigned Integer	None	
1351	NoApplIDs	Unsigned Integer	None	
1355	RefApplID	ASCII String	None	
1182	ApplBegSeqNum	Unsigned Integer	None	
1183	ApplEndSeqNum	Unsigned Integer	None	

6.4 Application Messages (Server-Initiated)

6.4.1 Security Definition

6.4.1.1 FIX Message

Tag	Field Name		Req	Description
35	MsgType		Y	d = Security Definition
52	SendingTime		Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID		Y	Identifier of the server sending the message.
1181	ApplSeqNum		N	Sequence number of the message on the Real-Time channel. Required if the message is disseminated via the Real-Time or Replay channel.
320	SecurityReqID		N	Identifier of the Security Definition Request this message relates to. Required if the message is disseminated via the Recovery channel.
323	SecurityResponse Type		N	Type of response. Required if the message is disseminated via the Recovery channel. **Value** **Meaning** 4 List of Securities Returned 6 Cannot Match Selection Criteria
912	LastRptRequested		N	Indicates the last message sent in response to a request. **Value** **Meaning** Y Last Message
48	SecurityID		N	Identification number for the security.
22	SecurityIdSource		N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier
965	SecurityStatus		N	Status of the instrument. **Value** **Meaning** 1 Active 9 Suspended
454	NoSecurityAltID		N	If present, value in this field will always be “1”.
➡	455	Security AltID	N	Identification number for the security.

➡	456	SecurityAltIDSource	N	Type of security identification number used. Required if SecurityAltID (455) is specified. **Value** **Meaning** 4 ISIN 8 Exchange Symbol M Market assigned identifier (TIDM)
423	PriceType		N	Quotation format for the instrument. Absence of this field should be interpreted as Price Per Unit (2). **Value** **Meaning** 2 Price Per Unit
1310	NoMarketSegments		N	Number of segments the instrument is assigned to. The value in this field will always be “1”.
➡	1300	MarketSegmentID	N	Segment the instrument is assigned to. Please refer to Section 7.1 for the valid segments.
292	CorporateAction		N	Gives the Ex-Marker and/or Annotation information (if any) associated with an instrument. Please refer to Section **Error! Reference source not found.** for valid Ex-Markers and Annotations.

6.4.1.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	Default	Please refer to Section 4.2.1
1181	ApplSeqNum	Unsigned Integer with NULL support	Increment	
320	SecurityReqID	ASCII String	Copy	
323	SecurityResponse Type	Unsigned Integer with NULL support	None	
912	LastRptRequested	ASCII String	Default	N
48	SecurityID	ASCII String	Copy	
22	SecurityIDSource	ASCII String	None	
965	SecurityStatus	ASCII String	Copy	
454	NoSecurityAltID	Unsigned Integer with NULL support	Default	3
455	SecurityAltID	ASCII String	Copy	
456	SecurityAltIDSource	ASCII String	Copy	
423	PriceType	Unsigned Integer	Copy	
1310	NoMarketSegments	Unsigned Integer	Constant	1
1300	Market SegmentID	ASCII String	Copy	
292	CorporateAction	ASCII String	Copy	

6.4.2 Security Status

6.4.2.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	f = Security Status
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID	Y	Identifier of the server sending the message.
1181	ApplSeqNum	Y	Sequence number of message on Real-Time channel. Required if message is disseminated via the Real-Time or Replay channel.
48	SecurityID	Y	Identification number for the security.
22	SecurityIDSource	N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier
326	SecurityTradingStatus	Y	Indicates the trading status of the instrument. **Value** **Meaning** 2 Halt 17 Regular Trading/Start Trade Reporting 18 Market Close 21 Opening Auction Call 26 Post-Close 100 Start of Trading 102 Closing Auction Call 103 End of Post Close 104 Volatility Auction Call 105 Re-Opening Auction Call 111 Pause 120 Closing Price Cross 121 Futures Close Out 122 Intra-Day Auction Call 125 Closing Price Publication 130 End Trade Reporting 135 EOD Volume Auction Call

<table>
<tr><td>1174</td><td>SecurityTradingEvent</td><td>N</td><td>Indicates the reason a trading session is extended or shortened.
Value Meaning
1 Market Order Imbalance
100 Price Outside Range
101 Session Extended
102 Session Shortened
103 Circuit Breaker Tripped</td></tr>
<tr><td>1173</td><td>MDSubBookType</td><td>N</td><td>Type of trading to which the update relates. Absence of this field should be interpreted as On Book (1).for all order books.
Value Meaning
1 On Book
2 Off Book</td></tr>
<tr><td>58</td><td>Text</td><td>N</td><td>Time at which the auction will take place will be specified in the case of a re-opening or the extension of a scheduled auction. The time the trading session will end in the case of a session extension or shortening. The time will be specified in the local time for the server (i.e. not in UTC) and in the HH:MM:SS format</td></tr>
<tr><td>327</td><td>HaltReason</td><td>N</td><td>Reason for the trading halt. Please refer to Section 8 for an explanation of the reason codes. Required if SecurityTradingStatus (326) is Halt (2).</td></tr>
</table>

6.4.2.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	Default	Please refer to Section 4.2.1
1181	ApplSeqNum	Unsigned Integer with NULL support	Increment	
48	SecurityID	ASCII String	Copy	
22	SecurityIDSource	ASCII String	Constant	8
326	SecurityTradingStatus	Unsigned Integer	Copy	
1174	SecurityTradingEvent	Unsigned Integer with NULL support	Copy	
1173	MDSubBookType	Unsigned Integer	Default	1
58	Text	ASCII String	Copy	
327	HaltReason	Unsigned Integer with NULL support	Copy	

6.4.3 Market Data Snapshot (Full Refresh)

6.4.3.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	W = Market Data Snapshot (Full Refresh)
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID	Y	Identifier of the server sending the message.
1181	ApplSeqNum	N	Sequence number of message on Real-Time channel. Required if message is disseminated via the Real-Time or Replay channel.
893	LastFragment	N	Indicates whether this message is the last in a sequence of snapshot messages for a given point of time **Value** **Meaning** Y Last Message N Not last Message
262	MDReqID	N	Identifier of the Market Data Request this message relates to. Required if the message is disseminated via the Recovery channel.
369	LastMsgSeqNum Processed	N	ApplSeqNum (1181) of the last incremental update on the Real-Time channel with which the snapshot is synchronised. Required if the message is disseminated via the Recovery channel.
912	LastRptRequested	N	Indicates the last message sent in response to a request. **Value** **Meaning** Y Last Message
48	SecurityID	Y	Identification number for the security.
22	SecurityIDSource	N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier
83	RptSeq	Y	Instrument specific sequence number of update in Real Time Channel. If the message is disseminated via the Recovery channel, this is the field which Identifies the instrument specific sequence number of the last incremental update with which the snapshot is synchronised.

<table>
<tr><td>1682</td><td>MDSecurityTrading Status</td><td>N</td><td>Indicates the trading status of the instrument. Required if the message is sent in the Recovery channel.

Value Meaning
2 Halt
17 Regular Trading/Start Trade Reporting
18 Market Close
21 Opening Auction Call
26 Post-Close
100 Start of Trading
102 Closing Auction Call
103 End of Post Close
104 Volatility Auction Call
105 Re-Opening Auction Call
111 Pause
120 Closing Price Cross
121 Futures Close Out
122 Intra-Day Auction Call
125 Closing Price Publication
130 End Trade Reporting
199 No Active Session
135 EOD Volume Auction Call</td></tr>
<tr><td>1684</td><td>MDHaltReason</td><td>N</td><td>Reason for the trading halt. Please refer to Section 8 for an explanation of the reason codes. Required if MDSecurityTradingStatus (1682) is Halt (2).</td></tr>
<tr><td>1173</td><td>MDSubBookType</td><td>N</td><td>Type of trading to which the update relates. Absence of this field should be interpreted as On Book (1).for all order books.
Value Meaning
1 On Book
2 Off Book</td></tr>
<tr><td>268</td><td>NoMDEntries</td><td>Y</td><td>Number of market data entries in the message.</td></tr>
</table>

<table>
<tr><td>➡</td><td>269</td><td>MDEntryType</td><td>Y</td><td>Indicates the type of market data being published. This will be the first field in the repeating group.
Value Meaning
0 Bid
1 Offer
4 Opening Price
5 Closing Price
7 High Price (For On Book trades only)
8 Low Price (For On Book trades only)
9 VWAP
B Volume
J Empty Order Book
b Market Bid
c Market Offer
d Turnover
e Number of Trades
f Previous Close
z No Statistics</td></tr>
<tr><td>➡</td><td>278</td><td>MDEntryID</td><td>N</td><td>Unique identifier of each order. Required if MDEntryType (269) is Bid (0), Offer (1), Market Bid (b) or Market Offer (c) and the order book is published by order depth.</td></tr>
<tr><td>➡</td><td>270</td><td>MDEntryPx</td><td>N</td><td>Price of the bid or offer being published. Required if MDEntryType (269) is not Volume (B), Market Bid (b) or Market Offer (c), number of trades (e).</td></tr>
<tr><td>➡</td><td>271</td><td>MDEntrySize</td><td>N</td><td>Quantity of the market data entry. Required if MDUpdateAction (279) is New (0) or Change (1) and MDEntryType (269) is not High Price (7), Low Price (8), VWAP (9), Turnover (d), PreviousClose (f).</td></tr>
<tr><td>➡</td><td>276</td><td>QuoteCondition</td><td>N</td><td>Value Meaning
O Closing

Will be used to indicate Closing Bid and Offer Price together with MDEntryType (269) of Bid (0) and Offer (1) respectively.</td></tr>
</table>

➡	346	NumberOfOrders		N	Number of orders represented in the aggregate quantity published for a bid or offer. Required if MDEntryType (269) is Bid (0), Offer (1), Market Bid (b) or Market Offer (c) and the order book is published by price depth.
➡	1023	MDPriceLevel		N	Display position of the price level in the order book. Required if MDEntryType (269) is Bid (0) or Offer (1).
➡	1024	MDOriginType		N	**Value** **Meaning** 0 Book 1 Off Book Required when MDEntryType (269) is Turnover (d). (0) indicates the statistic to be a Turnover (On Book) while (1) indicates Turnover (Off Book). Required when MDEntryType (269) is Volume (B). 0 indicates the statistic to be a Volume (On Book) while 1 indicates Volume (Off Book). Required when MDEntryType (269) is number of trades (e). 0 indicates the statistic to be number of trades (On Book) while 1 indicates number of trades (Off Book). MDEntryType (269) of VWAP (9) together with MDOriginType (1024) of Book (0) indicates VWAP (Automatic trades) statistic available in the system. MDEntryType (269) of VWAP (9) indicates VWAP of all trades.
➡	453	NoPartyIDs		N	Number of party identifiers. If specified, the value in this field will always be “1”.
➡	➡	448	PartyID	N	Identifier of the party. Required if NoPartyIDs (453) is specified.
➡	➡	447	PartyID Source	N	Required if PartyID (448) is specified. **Value** **Meaning** D Proprietary/Custom Code
➡	➡	452	Party Role	N	Role of the specified PartyID (448). Required if PartyID (448) is specified. **Value** **Meaning** 1 Trading Firm

<table>
<tr>
<td>➡</td>
<td>30002</td>
<td>OpenClose Indicator</td>
<td>N</td>
<td>Method used to compute the opening or closing price. Required if MDEntryType (269) is Opening Price (4) or Closing Price (5).
If the MDEntryType (269) is Opening Price (4) the value indicated in this field will be the opening price indicator.
If the MDEntryType (269) is Closing Price (5) the value indicated in this field will be the closing price indicator.

<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>1</td><td>UT</td></tr>
<tr><td>2</td><td>AT</td></tr>
<tr><td>3</td><td>Mid of BBO after opening auction</td></tr>
<tr><td>4</td><td>Mid of BBO before closing auction call</td></tr>
<tr><td>6</td><td>VWAP</td></tr>
<tr><td>7</td><td>Last AT</td></tr>
<tr><td>8</td><td>Last UT</td></tr>
<tr><td>9</td><td>Previous Close</td></tr>
<tr><td>10</td><td>Manual</td></tr>
<tr><td>21</td><td>Zero</td></tr>
<tr><td>22</td><td>Best Bid</td></tr>
<tr><td>23</td><td>Best Offer</td></tr>
</table>
</td>
</tr>
</table>

6.4.3.2 FAST Template

Tag	Field Name	Data Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	Default	Please refer to Section 4.2.1
1181	ApplSeqNum	Unsigned Integer with NULL support	Increment	
893	LastFragment	ASCII String	Copy	Y
262	MDReqID	ASCII String	Copy	
369	LastMsgSeqNum Processed	Unsigned Integer	None	
912	LastRptRequested	ASCII String	Default	N
48	SecurityID	ASCII String	Copy	
22	SecurityIDSource	ASCII String	Constant	8
83	RptSeq	Unsigned Integer with NULL support	None	
1682	MDSecurity TradingStatus	Unsigned Integer with NULL support	Copy	
1684	MDHaltReason	Unsigned Integer with NULL support	Copy	
1173	MDSubBookType	Unsigned Integer	Default	1
268	NoMDEntries	Unsigned Integer	Default	1
269	MDEntryType	ASCII String	Copy	
278	MDEntryID	ASCII String	Copy	
270	MDEntryPx	Scaled Number	Copy	
271	MDEntrySize	Scaled Number	Copy	
276	QuoteCondition	ASCII String	Copy	
346	NumberOfOrders	Unsigned Integer with NULL support	Copy	
1023	MDPriceLevel	Unsigned Integer	Copy	
1024	MDOriginType	Unsigned Integer with NULL support	Copy	

453	NoPartyIDs	Unsigned Integer with NULL support	Constant	1
448	PartyID	ASCII String	Copy	
447	PartyIDSource	ASCII String	Copy	
452	PartyRole	Unsigned Integer with NULL support	Copy	
30002	OpenClose Indicator	Unsigned Integer with NULL support	Copy	

6.4.4 Market Data Incremental Refresh

6.4.4.1 FIX Message

Tag	Field Name		Req	Description
35	MsgType		Y	X = Market Data - Incremental Refresh
52	SendingTime		Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID		Y	Identifier of the server sending the message.
1181	ApplSeqNum		N	Sequence number of message on the Real-Time channel.
262	MDReqID		N	Identifier of the Market Data Request this message relates to. Required if the message is disseminated via the Snapshot channel.
912	LastRptRequested		N	Indicates the last message sent in response to a retransmission request. **Value** **Meaning** Y Last Message
268	NoMDEntries		Y	Number of market data entries in the message.
➡	279	MDUpdate Action	Y	Indicates the update type. **Value** **Meaning** 0 New 1 Change 2 Delete
➡	1173	MDSubBookType	N	Type of trading to which the update relates. **Value** **Meaning** 1 Regular 2 Off-Book

➡	269	MDEntryType	Y	Indicates the type of market data being published. **Value** **Meaning** 0 Bid 1 Offer 2 Trade 4 Opening Price 5 Closing Price 7 High Price (For On Book trades only) 8 Low Price (For On Book trades only) 9 VWAP B Volume Q Auction Clearing Price d Turnover e Number of Trades f Previous Close b Market Bid c Market Offer
➡	278	MDEntryID	N	Unique identifier of a market data entry. Required if MDEntryType (269) is Bid (0), Offer (1), Market Bid (b) or Market Offer (c). Required if MDEntryType (269) is Trade (2).
➡	48	SecurityID	Y	Identification number for the security.
➡	22	SecurityIDSource	N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier
➡	270	MDEntryPx	N	Price of the market data entry. Required if MDUpdateAction (279) is New (0) or Change (1) and if MDEntryType (269) is not Volume (B), Market Bid (b) or Market Offer (c), number of trades (e).
➡	271	MDEntrySize	N	Quantity of the market data entry. Required if MDUpdateAction (279) is New (0) or Change (1) and MDEntryType (269) is not High Price (7), Low Price (8), VWAP (9), Turnover (d), PreviousClose (f), Opening Price (4) and Closing price (5).
➡	272	MDEntryDate	N	The Date on which an Off Book trade was agreed between the firms. This will be required only when the MDEntryType (269) is Trade (2). . The date will be specified in UTC and in the YYYYMMDD format. If no value is tagged it is considered to be the current trading date.

<table>
<tr><td>➡</td><td>273</td><td>MDEntryTime</td><td>N</td><td>Time the On Book trade being published was executed.
The time an Off Book trade was agreed between the firms will be specified in this field.
The time will be specified in UTC and in the HH:MM:SS.sss format.</td></tr>
<tr><td>➡</td><td>276</td><td>QuoteCondition</td><td>N</td><td><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>O</td><td>Closing</td></tr></table>
Will be used to indicate the Bid and Offer Price used to determine the Closing Price together with MDEntryType (269) of Bid (0) and Offer (1) respectively.</td></tr>
<tr><td>➡</td><td>277</td><td>Trade Condition</td><td>N</td><td>.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>R</td><td>Opening Price</td></tr><tr><td>w</td><td>Volatility</td></tr><tr><td>AJ</td><td>Closing Price</td></tr><tr><td>AZ</td><td>Re-Opening Auction</td></tr><tr><td>7</td><td>Intra-Day Auction</td></tr><tr><td>FC</td><td>Futures Close Out Auction</td></tr><tr><td>VU</td><td>EOD Volume Auction</td></tr></table>Required when MDEntryType (269) is Trade (2) and if the Off Book trade reported is a late trade.
Trade time indicator - Late Trade is indicated by value (M).
Required if MDEntryType (269) is Trade (2) and MatchType (574) is Auction (5) to indicate the auction indicator. R, w, AJ, AZ, 7, VU and FC will be the auction indicators.</td></tr>
<tr><td>➡</td><td>574</td><td>MatchType</td><td>N</td><td>Whether a trade was executed in an auction.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>5</td><td>Auction</td></tr></table></td></tr>
<tr><td>➡</td><td>346</td><td>Number OfOrders</td><td>N</td><td>Number of orders represented in the aggregate quantity published for a bid or offer. Required if MDUpdateAction (279) is New (0) or Change (1), MDEntryType (269) is Bid (0), Offer (1), Market Bid (b) or Market Offer (c) and the order book is published by price depth.</td></tr>
<tr><td>➡</td><td>1023</td><td>MDPriceLevel</td><td>N</td><td>Display position of a price level in the order book. Required if MDEntryType (269) is Bid (0) or Offer (1).</td></tr>
</table>

➡	1024	MDOriginType		N	**Value** **Meaning** 0 Book 1 Off Book Required when MDEntryType (269) is Turnover (d). 0 indicates the statistic to be a Turnover (On Book) while 1 indicates Turnover (Off Book). Required when MDEntryType (269) is Volume (B). 0 indicates the statistic to be a Volume (On Book) while 1 indicates Volume (Off Book). Required when MDEntryType (269) is number of trades (e). 0 indicates the statistic to be number of trades (On Book) while 1 indicates number of trades (Off Book). MDEntryType (269) of VWAP (9) together with MDOriginType (1024) of Book (0) indicates VWAP (Automatic trades) statistic available in the system. MDEntryType (269) of VWAP (9) indicates VWAP of all trades.
➡	1070	MDQuote Type		N	**Value** **Meaning** 0 Indicative Indicative (0) required if MDEntryType (269) is Auction Clearing Price (Q).
➡	83	RptSeq		N	Instrument specific sequence number of update.
➡	453	NoPartyIDs		N	Number of party identifiers. If specified, the value in this field will always be “1”.
➡	➡	448	PartyID	N	Identifier of the party. Required if NoPartyIDs (453) is specified.
➡	➡	447	PartyID Source	N	Required if PartyID (448) is specified. **Value** **Meaning** D Proprietary/Custom Code
➡	➡	452	Party Role	N	Role of the specified PartyID (448). Required if PartyID (448) is specified. **Value** **Meaning** 1 Trading Firm
➡	829	TrdSubType		N	Required when the MDEntryType (269) is Trade (2) Indicates the type of the trade disseminated, Please refer to Section 10 for the TrdSubTypes and the relevant values to fill this field. This field will contain the value 1016 - XT for cross trades

➡	30002	OpenClose Indicator	N	Method used to compute the opening or closing price. Required when the MDEntryType (269) is Opening Price (4) or Closing Price (5). If the MDEntryType (269) is Opening Price (4) the value indicated in this field will be the opening price indicator. If the MDEntryType (269) is Closing Price (5) the value indicated in this field will be the closing price indicator.

Value	Meaning
1	UT
2	AT
3	Mid of BBO after opening auction
4	Mid of BBO before closing auction call
6	VWAP
7	Last AT
8	Last UT
9	Previous Close
10	Manual
21	Zero
22	Best Bid
23	Best Offer

6.4.4.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	Default	Please refer to Section 4.2.1
1181	ApplSeqNum	Unsigned Integer with NULL support	Increment	
262	MDReqID	ASCII String	Copy	
912	LastRptRequested	ASCII String	Default	N
268	NoMDEntries	Unsigned Integer	Default	1
279	MDUpdateAction	Unsigned Integer	Copy	
1173	MDSubBookType	Unsigned Integer	Copy	

269	MDEntryType	ASCII String	Copy	
278	MDEntryID	ASCII String	Copy	
48	SecurityID	ASCII String	Copy	
22	SecurityIDSource	ASCII String	Constant	8
270	MDEntryPx	Scaled Number	Copy	
271	MDEntrySize	Scaled Number	Copy	
272	MDEntryDate	ASCII String	Tail	
273	MDEntryTime	ASCII String	Tail	
276	QuoteCondition	ASCII String	Copy	
277	TradeCondition	ASCII String	Copy	
574	MatchType	ASCII String	Copy	
346	NumberOfOrders	Unsigned Integer with NULL support	Copy	
1023	MDPriceLevel	Unsigned Integer	Copy	
1024	MDOriginType	Unsigned Integer with NULL support	Copy	
1070	MDQuoteType	Unsigned Integer with NULL support	Copy	
83	RptSeq	Unsigned Integer with NULL support	Increment	
453	NoPartyIDs	Unsigned Integer with NULL support	Constant	1
448	PartyID	ASCII String	Copy	
447	PartyIDSource	ASCII String	Copy	
452	PartyRole	Unsigned Integer with NULL support	Copy	
829	TrdSubType	Unsigned Integer with NULL support	Copy	

30002	OpenClose Indicator	Unsigned Integer with NULL support	Copy	

6.4.5 News

This message is used to disseminate Market Operation announcements to the market and is not related to regulatory News announcements.

6.4.5.1 FIX Message

Tag	Field Name		Req	Description
35	MsgType		Y	B = News
52	SendingTime		Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1180	ApplID		Y	Identifier of the server sending the message.
1181	ApplSeqNum		N	Sequence number of the message on the Real-Time channel. Required if the message is disseminated via the Real-Time or Replay channel.
42	OrigTime		Y	Time the announcement was published which will be specified in UTC and in the HH:MM:SS format.
61	Urgency		Y	Level of urgency of the announcement. **Value** **Meaning** 0 Normal 1 Flash (High Priority) 2 Background (Low Priority)
148	Headline		Y	Headline or subject of the announcement.
912	LastRptRequested		N	Indicates the last message sent in response to a request. **Value** **Meaning** Y Last Message
33	NoLinesOfText		Y	Number of lines of text. The value in this field will always be “1”.
➡	58	Text	Y	Text of the announcement.
146	NoRelatedSym		N	Number of related instruments.
➡	48	SecurityID	N	Identification number for the security.
➡	22	SecurityIDSource	N	Type of security identification number used. Required if SecurityID (48) is specified. **Value** **Meaning** 8 instrument identifier

6.4.5.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1180	ApplID	ASCII String	Default	Please refer to Section 4.2.1
1181	ApplSeqNum	Unsigned Integer with NULL support	Increment	
42	OrigTime	ASCII String	Tail	
61	Urgency	Unsigned Integer	Default	0
148	Headline	ASCII String	None	
912	LastRptRequested	ASCII String	Copy	
33	NoLinesOfText	Unsigned Integer	Constant	1
58	Text	ASCII String	None	
146	NoRelatedSym	Unsigned Integer with NULL support	Default	1
48	SecurityID	ASCII String	Copy	
22	SecurityIDSource	ASCII String	Constant	8

6.4.6 Market Data Request Reject

6.4.6.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	Y = Market Data Request Reject
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
262	MDReqID	Y	Identifier of the request being rejected.
281	MDReqRej Reason	Y	Code specifying the reason for the rejection. Please refer to Section 9.1 for a list of reject codes.
58	Text	N	JSE specific code specifying the reason for the reject. Please refer to Section 9.1 for a list of reject codes.

6.4.6.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
262	MDReqID	ASCII String	None	
281	MDReqRejReason	ASCII String	None	
58	Text	ASCII String	None	

6.4.7 Business Message Reject

6.4.7.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	j = Business Message Reject
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
379	BusinessReject RefID	N	SecurityReqID (320), MDReqID (262) or ApplReqID (1346) 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.
372	RefMsgType	Y	MsgType (35) of the rejected message.
380	BusinessReject Reason	Y	Code specifying the reason for the reject. Please refer to Section 9.2 for a list of reject codes.
58	Text	N	JSE specific code specifying the reason for the reject. Please refer to Section 9.2 for a list of reject codes.

6.4.7.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
379	BusinessReject RefID	ASCII String	None	
371	RefTagID	Unsigned Integer with NULL support	None	
372	RefMsgType	ASCII String	None	
380	BusinessReject Reason	Unsigned Integer with NULL support	None	
58	Text	ASCII String	None	

6.4.8 Application Message Request Ack

6.4.8.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	BX = Application Message Request Ack
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1353	ApplResponseID	Y	Server specified identifier of the acknowledgement.
1346	ApplReqID	Y	Identifier of the request being acknowledged.
1347	ApplReqType	Y	Type of request being acknowledged. **Value** **Meaning** 0 Retransmission of Messages 5 Cancel Retransmission
1348	ApplResponse Type	Y	Whether the request was successful. **Value** **Meaning** 0 Request Successful 1 Unknown ApplID 2 Messages Not Available

6.4.8.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1353	ApplResponseID	ASCII String	None	
1346	ApplReqID	ASCII String	None	
1347	ApplReqType	Unsigned Integer	None	
1348	ApplResponse Type	Unsigned Integer	None	

6.4.9 Application Message Report

6.4.9.1 FIX Message

Tag	Field Name	Req	Description
35	MsgType	Y	BY = Application Message Report
52	SendingTime	Y	Time the gateway server sent the message specified in UTC and in the YYYYMMDD-HH:MM:SS.sss format.
1356	ApplReportID	Y	Server specified identifier of the report.
1346	ApplReqID	Y	Identifier of the Application Message Request the report relates to.

1426	ApplReportType	Y	**Value**	**Meaning**
			3	Retransmission Completed

6.4.9.2 FAST Template

Tag	Field Name	Field Type	Field Encoding	Value
35	MsgType	ASCII String	None	
52	SendingTime	ASCII String	None	
1356	ApplReportID	ASCII String	None	
1346	ApplReqID	ASCII String	None	
1426	ApplReportType	Unsigned Integer	None	

7 INSTRUMENT CATEGORIES

7.1 Segment

Segment	Description
ZA01	JSE Top Companies
ZA02	JSE Medium Liquid
ZA03	JSE Less Liquid
ZA04	JSE Specialist Products
ZA06	JSE Exchange Traded Products
ZA11	NSX Local Listed Companies
ZA12	NSX Dual Listed Companies

8 TRADING HALT REASON CODES

Code	Reason
9998	Matching partition suspended
9999	System suspended
1	System Issues Being Experienced
2	Company Announcement Expected
3	Company Requested Halt
4	Company Requested Suspension
5	JSE Initiated Halt/ Suspension
102	Instrument Status is Halted
Space	Reason not available

9 REJECT CODES

9.1 Market Data Request Reject

MDReq Rej Reason	Text	Reason
0	-	Unknown instrument
4	-	Unsupported SubscriptionRequestType
8	-	Unsupported MDEntryType
Z	-	Other
Z	101	Unknown segment
Z	102	Requested market data unavailable for instrument

9.2 Business Message Reject

Business Reject Reason	Text	Reason
0	400	Other
0	403	Incorrect data format for this tag
0	404	Value is invalid for this tag
0	450	Request limit for day reached
1	-	Unknown ID
3	-	Unsupported message type
5	-	Conditionally required field missing

10 VALID TRADE TYPES

~~10.1 TrdSubType~~

10.1 Off Book Trade Types

TrdSubType (829) Value	Description
17	LC – Off Book Post Contra Trade (Cancellation of previous day's published Off Book trade)
24	PC – On Book Post Contra Trade (Cancellation of previous day's On Book trade)
2001	BT - Block Trade
2002	CF - Corporate Finance Trade
2003	LT - Late Trade, (After Hours Trade)
2004	NX - Namibia Trade
2005	OD - Delta Trade
2006	OP - Off Order Book Principal Trade
2007	OX - Option Exercised
2008	TX - Traded Option Exercised
2009	PF - Portfolio Trade
2011	WX - Warrant Exercised
2013	GU – Give Up Trade
3001	BK – Book Build
3015	NC – Off Book Post Contra Trade Non-published (Cancellation of previous day's non-published Off Book trade) This trade type will never be published via market data, it is only valid for the Post Trade Gateway when cancelling a non-published trade.
1016	XT – Cross Trade

11 CORPORATE ACTION INFORMATION

11.1 Valid Ex-Markers

Ex-Marker Code	Description
GT	Green Triangle – Declared Payment (dividend or other type of payment) still to be paid
XD	Ex-Dividend or other payment

11.2 Valid Annotations

Annotation Code	Description
A^	Adverse Auditors Opinion Expressed
D^	Disclaimed Annual Audit Opinion
E^	Annual Audit Report “Emphasis of matter” paragraph
OT	Orange Triangle – Caution in dealing in shares
Q^	Qualified Annual Audit Opinion
R^	Failure to Provide Annual Compliance Certificate
RE	Red Square – Company Violation of the JSE Rules
SV	Shareholder Spread Violation
TA	Tax Applicable

Note: The Exchange may amend these values at any time following one month’s prior written notice to clients. Single characters on Ex-Markers and Annotations will be right padded with a space as denoted above with the ‘^’ symbol.

12 MARKET ~~DATA~~DATA GATEWAYS

1 Service ~~Diagrams~~refers to a single market data gateway partition instance. i.e. primary and mirror services (also known as A Feed and B Feed) hosted as primary and mirror gateway instances.

Protocol	Markets	No of Un-throttled GWs	Un-throttled services	No of Throttled GW	Throttled services
FAST	EQM	0	-	6	1. Level 1 [Book Depth 1] – JSE EQM 2. Level 1 [Book Depth 1] – NSX 3. Indices Feed– JSE EQM 4. Indices Feed- NSX 5. SENS Feed – JSE EQM 6. SENS Feed – NSX