S&P Capital IQ's Real-Time Solutions

QuantFEED® Feed Description

JSE MIT Feed

Reference n°: 20131114

S&P Capital IQ's Real-Time Solutions (QuantHouse®) – QuantFEED®
QuantFEED® Feed Description
Reference 20131114
November 14, 2013

Corporate Headquarters
S&P Capital IQ's Real-Time Solutions (QuantHouse®)
52 Rue de la Victoire
75009 Paris
France
Tel: +33 (0) 1 73 02 32 11
Fax: +33 (0) 1 73 02 32 12

UK Office
10 Foster Lane
London EC2V 6HR
United Kingdom
Tel: +44 (0) 203 107 1676

US Offices
55 Water Street, 44th floor
New York, NY 10041
United States of America
Tel: +1-(212)-438-4346

130 East Randolph
One Prudential Plaza, Suite 2900
Chicago, IL 60601
United States of America
Tel: +1-(312)-233-7129

www.quanthouse.com

TABLE OF CONTENTS

QUANTFEED® JSE MIT FEED DESCRIPTION

As part of S&P Capital IQ's Real-Time Solutions's QuantFEED® documentation, this feed description provides you with details about the types of data broadcast on the JSE MIT market data stream, their possible values and current QuantFEED® technical implementation.

The topics this feed description covers include*:

1. Referential Data

The following sections describe the characteristics of the referential data on JSE MIT market data stream, in terms of:

1.1. Available Markets and Branches

This section details the list of markets and branches available on JSE MIT market data stream:

* The red bars in the left margin highlight content that has been added or changed since the previous release of this document.

1.1.1. Markets

The JSE MIT market data stream broadcasts informations about the following markets:

Table 1 List of markets available on JSE MIT market data stream

QuantFEED® Market ID	Market
XJSE	JSE Securities Exchange

The following example shows the list of markets available on JSE MIT market data stream and their IDs, returned by the command `dumps`:

```
MARKETS
market # 232    CC=ZA/SOUTH AFRICA/JOHANNESBURG,DESCR=JSE SECURITIES EXCHANGE,
WEB=www.jse.co.za
    MIC = XJSE
    TimeZone = Africa/Johannesburg
    Country = ZA
    NbMaxInstruments = 2000000
```

1.1.2. Branches

The example below shows the list of branches available on JSE MIT market data stream, returned by the command `dumps`. Each branch displays the following details: `FOSMarketID`, `SecurityType`, `CFICode` and `Quantity` (of instruments):

```
BRANCHES
    { XJSE CS   ESXXXX } qty: 407
    { XJSE NONE DBXXXX } qty: 4
    { XJSE NONE EUXXXX } qty: 38
    { XJSE NONE MRCXXX } qty: 4
    { XJSE NONE XXXXXX } qty: 426
    { XJSE PS   ERXXXX } qty: 45
    { XJSE WAR  RWXXXX } qty: 103
```

1.2. Types of Instruments

The following sections illustrate the instruments' characteristics on JSE MIT market data stream, according to their type:

- 1.2.1. Equities
- 1.2.2. Bonds
- 1.2.3. Warrants
- 1.2.4. Miscellaneous.

1.2.1. Equities

The sample below illustrates the details of an equity:

```
instr # 232/1001 = 486540265
    PriceCurrency               string{ZAC}
    Symbol                      string{BTI}
    Description                 string{British American Tob plc}
    SecurityType                string{CS}
    FOSMarketId                 XJSE
    CFICode                     string{ESXXXX}
    RoundLot                    float64{1}
    SecurityGroup               string{ZA01}
    InternalCreationDate        Timestamp{2012-06-30 23:37:32:068}
    InternalModificationDate    Timestamp{2012-10-09 23:00:13:256}
    InternalSourceId            uint16{31}
    LocalCodeStr                string{12511}
    ISIN                        string{GB0002875804}
    PriceIncrement_static       float64{1}
    OperatingMIC                string{XJSE}
    MARKET_LSE_NormalMarketSize float64{35000}
    MARKET_LSE_SectorCode       string{J1H1}
```

1.2.2. Bonds

The sample below illustrates the details of a bond:

```
instr # 232/3648 = 486542912
    PriceCurrency               string{ZAC}
    Symbol                      string{SHPCB}
    Description                 string{Shoprite Investments Limited}
    SecurityType                string{NONE}
    FOSMarketId                 XJSE
    CFICode                     string{DBXXXX}
    RoundLot                    float64{1}
    SecurityGroup               string{ZA02}
    InternalCreationDate        Timestamp{2012-06-30 23:37:32:119}
    InternalModificationDate    Timestamp{2012-10-09 23:00:13:281}
    InternalSourceId            uint16{31}
    LocalCodeStr                string{29884}
    ISIN                        string{ZAE000166997}
    PriceIncrement_static       float64{1}
    OperatingMIC                string{XJSE}
    MARKET_LSE_NormalMarketSize float64{5000}
    MARKET_LSE_SectorCode       string{J2H1}
```

1.2.3. Warrants

The sample below illustrates the details of a warrant:

```
instr # 232/2232 = 486541496
    PriceCurrency               string{ZAC}
    Symbol                      string{SYCIVC}
    Description                 string{IB SYC 663BC 1:1JUL19}
    SecurityType                string{WAR}
    FOSMarketId                 XJSE
    CFICode                     string{RWXXXX}
    RoundLot                    float64{1}
    SecurityGroup               string{ZA04}
    InternalCreationDate        Timestamp{2012-06-30 23:37:32:070}
    InternalModificationDate    Timestamp{2012-10-09 23:00:13:258}
    InternalSourceId            uint16{31}
    LocalCodeStr                string{13390}
    ISIN                        string{ZAE000138228}
    PriceIncrement_static       float64{1}
    OperatingMIC                string{XJSE}
    MARKET_LSE_NormalMarketSize float64{5000}
    MARKET_LSE_SectorCode       string{J4S2}
```

1.2.4. Miscellaneous

The sample below illustrates the details of a miscellaneous instrument:

```
instr # 232/1894 = 486541158
    PriceCurrency               string{ZAC}
    Symbol                      string{KRQT}
    Description                 string{Kruger Rand Quarter}
    SecurityType                string{NONE}
    FOSMarketId                 XJSE
    CFICode                     string{MRCXXX}
    RoundLot                    float64{1}
    SecurityGroup               string{ZA02}
    InternalCreationDate        Timestamp{2012-06-30 23:37:32:083}
    InternalModificationDate    Timestamp{2012-10-09 23:00:13:264}
    InternalSourceId            uint16{31}
    LocalCodeStr                string{176}
    ISIN                        string{ZAE000003976}
    PriceIncrement_static       float64{1}
    OperatingMIC                string{XJSE}
    MARKET_LSE_NormalMarketSize float64{5000}
    MARKET_LSE_SectorCode       string{J2H1}
```

1.3. Specific Referential Tags

The following sections describe additional, specific referential tags available on JSE MIT market data stream:

- 1.3.3. Sector Code.

1.3.1. Operating MIC

The values of the referential tag **Operating MIC** conveyed on the JSE MIT market data stream are disseminated via QuantFEED®'s data stream in *Referential* to specify the parent MIC.

QuantFEED®'s implementation of the values currently available for the tag `OperatingMIC` is described in the table below:

Table 2 OperatingMIC – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`OperatingMIC`	QuantFEED® tag name.
Numeric ID	`9533`	QuantFEED® unique ID disseminated on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`String`	String data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, specifying the parent MIC.
Possible Values	`XJSE`	Parent MIC for all JSE's branches.

1.3.2. Normal Market Size

The referential tag **Normal Market Size** is disseminated via S&P Capital IQ's Real-Time Solutions's market data stream in *Referential* to detail the size of the transactions.

QuantFEED®'s implementation of the tag `MARKET_LSE_NormalMarketSize` is described in the following table:

Table 3 MARKET_LSE_NormalMarketSize – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_LSE_NormalMarketSize`	QuantFEED® tag name.
Numeric ID	`11000`	QuantFEED® unique ID disseminated on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`Float64`	Float64 data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the size of the transactions.

1.3.3. Sector Code

The referential tag **Sector Code** is disseminated via S&P Capital IQ's Real-Time Solutions's market data stream in *Referential* to identify a division of the market within a Market Segment.

QuantFEED®'s implementation of the tag `MARKET_LSE_SectorCode` is described in the following table:

Table 4 MARKET_LSE_SectorCode – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_LSE_SectorCode`	QuantFEED® tag name.
Numeric ID	`11001`	QuantFEED® unique ID disseminated on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`String`	String data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating a division of the market within a Market Segment.
Possible Values	`N1M2`	Namibian market local NSX mid-priced
	`N1L2`	Namibian market local NSX low-priced
	`J4S2`	South African market specialist products (IPs)
	`N1H2`	Namibian market local NSX high-priced
	`J1H1`	South African market dual-listed UK high-priced
	`J1M2`	South African market Top Companies mid-priced
	`J1M1`	South African market dual-listed UK mid-priced
	`J1H2`	South African market Top Companies high-priced
	`N1M2`	Namibian market local NSX mid-priced
	`NA01`	JSE/Namibia Listed

Table 4 MARKET_LSE_SectorCode – technical implementation in QuantFEED® (Continued)

Component	Value	Description
Possible Values	J2H2	South African market medium liquid high-priced
	J3L1	South African market less liquid low-priced
	J2H1	South African market high liquid high-priced
	J3H1	South African market less liquid high-priced
	J3M1	South African market less liquid mid-priced
	J2L1	South African market high liquid low-priced
	ALT	South African Market Alternative Exchange
	J4S1	South African market specialist products (Warrants)
	J2M1	South African market high liquid mid-priced
	J2L2	South African market medium liquid low-priced
	J2M2	South African market medium liquid mid-priced
	N2M1	Dual Listed NSX Mid Priced
	N2H1	Dual Listed NSX High Priced
	N2L1	Dual Listed NSX Low Priced
	ETF	Exchange Traded Funds
	BEE	Black Economic Empowerment
	BWO1	JSE/Botswana Listed

2. Quotation Data

The following sections describe the characteristics of the quotation data on JSE MIT market data stream, in terms of:

- 2.1. Quotation Values
- 2.2. Trading Status
- 2.3. Specific Quotation Tags.

2.1. Quotation Values

The examples below shows the possible values of an instrument on JSE MIT market data stream:

```
InstrumentStatusL1
-- 232/1001
        BID: 44713      500     @1
        ASK: 44750      400     @1
        LastPrice                       float64{44751}
        LastTradeQty                    float64{11}
        DailyHighPrice                  float64{45034}
        DailyLowPrice                   float64{44400}
        DailyTotalVolumeTraded          float64{403197}
        DailyTotalAssetTraded           float64{18009381037}
        LastTradePrice                  float64{44751}
        LastTradeTimestamp              Timestamp{2012-10-10 13:23:46:381}
        InternalDailyOpenTimestamp      Timestamp{2012-10-10 07:00:09:176}
        InternalDailyCloseTimestamp     Timestamp{2012-10-09 15:00:26:355}
        InternalDailyHighTimestamp      Timestamp{2012-10-10 10:10:43:755}
        InternalDailyLowTimestamp       Timestamp{2012-10-10 07:18:03:891}
        InternalPriceActivityTimestamp  Timestamp{2012-10-10 13:28:31:086}
        TradingStatus                   17=ReadyToTrade
        LastOffBookTradePrice           float64{44800}
        LastOffBookTradeQty             float64{21511}
        LastOffBookTradeTimestamp       Timestamp{2012-10-10 10:14:09:314}
        DailyOpeningPrice               float64{44800}
        PreviousDailyTotalVolumeTraded  float64{415843}
        PreviousDailyTotalAssetTraded   float64{18800336825}
        PreviousDailyClosingPrice       float64{45059}
        PreviousBusinessDay             Timestamp{2012-10-09}
        CurrentBusinessDay              Timestamp{2012-10-10}
        LastAuctionPrice                float64{44800}
        LastAuctionVolume               float64{725}
        DailyTotalOffBookVolumeTraded   float64{112304}
        DailyTotalOffBookAssetTraded    float64{5015701393}
        InternalLastAuctionTimestamp    Timestamp{2012-10-10 07:00:07:907}
        MARKET_LSE_SuspendedIndicator   char{N}
        MARKET_LSE_MIT_TradingStatusDetails     char{T}
        MARKET_JSE_MIT_TradingStatusDetails     char{T}
```

For more details about the fields and tags available in quotation data type, and their possible values, see *QuantFEED® Quotation Tags Guide*.

2.2. Trading Status

Each time a modification of the trading status occurs, the values of the quotation tag **Trading Status** conveyed on the JSE MIT market data stream are disseminated via QuantFEED®'s data stream in *Other Values*:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED®'s implementation of the tag `TradingStatus` is described in the following table:

Table 5 TradingStatus – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`TradingStatus`	QuantFEED® tag name.
Numeric ID	9100	QuantFEED® unique ID disseminated on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`Enum`	Enum data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the characteristics of the trading status.
Possible Values	2	Trading Halt
	3	Resume
	5	Price Indication
	17	Ready to Trade
	18	Not Available for Trading
	20	Unknown
	21	Pre-open

2.3. Specific Quotation Tags

The following sections describe additional, specific quotation tags available on JSE MIT market data stream:

- 2.3.1. Trade Conditions
- 2.3.2. Other Values.

2.3.1. Trade Conditions

The following subsections describe the trade conditions on JSE MIT market data stream:

- 2.3.1.1. Off Book Reporting Trade Type Indicator
- 2.3.1.2. Auction Type Indicator.

2.3.1.1. Off Book Reporting Trade Type Indicator

The values of the quotation tag **Off Book Reporting Trade Type Indicator** conveyed on the JSE MIT market data stream are disseminated via S&P Capital IQ's Real-Time Solutions's data stream in *Context* to detail the off book trade type:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED® implementation of the tag `MARKET_LSE_MIT_OffBookReportingTradeTypeIndicator` is described in the table below:

Table 6 MARKET_LSE_MIT_OffBookReportingTradeTypeIndicator – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_LSE_MIT_OffBookReportingTradeTypeIndicator`	QuantFEED® tag name.
Numeric ID	`15950`	QuantFEED® unique ID broadcast on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`String`	String data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the off book trade type.
Possible Values	`17`	LC – Cancellation of Previous Day's published Off Book Trade
	`24`	PC – Cancellation of previous day's On Book Trade
	`2001`	BT – Block Trade
	`2002`	CF – Corporate Finance Trade
	`2003`	LT – Late Trade
	`2004`	NX – Namibia Trade
	`2005`	OD – Delta Trade
	`2006`	OP – Off Order Book Principal Trade
	`2007`	OX – Option
	`2008`	TX – Traded Option Exercised
	`2009`	PF – Portfolio Trade
	`2011`	WX – Warrant
	`2013`	GU – Give Up
	`3001`	BK – Book Build
	`3015`	NC – Cancellation of previous day's non-published Off Book Trade. **Note:** This trade type will never be published via the market data, it is only valid via the Post Trade Gateway when cancelling a non-published trade.

2.3.1.2. Auction Type Indicator

The values of the quotation tag **Auction Type Indicator** conveyed on the JSE MIT market data stream are disseminated via S&P Capital IQ's Real-Time Solutions's data stream in *Context* to detail the auction type:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED® implementation of the tag `MARKET_JSE_MIT_AuctionTypeIndicator` is described in the table below:

Table 7 MARKET_JSE_MIT_AuctionTypeIndicator – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_JSE_MIT_AuctionTypeIndicator`	QuantFEED® tag name.
Numeric ID	`16320`	QuantFEED® unique ID broadcast on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the auction type.
Possible Values	`C`	Closing Auction
	`A`	Volatility
	`O`	Opening Auction
	`E`	Re-Opening Auction
	`K`	Intra-Day Auction
	`L`	Futures Closeout Auction

2.3.2. Other Values

The following subsections describe the trade conditions on JSE MIT market data stream:

- 2.3.2.1. Suspended Indicator
- 2.3.2.2. Halt Reason
- 2.3.2.3. Trading Status Details.

2.3.2.1. Suspended Indicator

Each time an instrument is suspended, the values of the quotation tag **Suspended Indicator** conveyed on the JSE MIT market data stream are disseminated via S&P Capital IQ's Real-Time Solutions's data stream in *Other Values* to indicate the status of the instrument:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED® implementation of the tag `MARKET_LSE_SuspendedIndicator` is described in the table below:

Table 8 MARKET_LSE_SuspendedIndicator – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_LSE_SuspendedIndicator`	QuantFEED® tag name.
Numeric ID	`14602`	QuantFEED® unique ID broadcast on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.

Table 8 MARKET_LSE_SuspendedIndicator – technical implementation in QuantFEED® (Continued)

Component	Value	Description
Type	`String`	String data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating the status of a suspended instrument.
Possible Values	`S`	Suspended Instrument
	`[Not sent] or Space`	Active Instrument

2.3.2.2. Halt Reason

Each time an instrument is halted from trading, the values of the quotation tag **Halt Reason** conveyed on the JSE MIT market data stream are disseminated via S&P Capital IQ's Real-Time Solutions's data stream in *Other Values*:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED® implementation of the tag `MARKET_LSE_MIT_HaltReason` is described in the table below:

Table 9 MARKET_LSE_MIT_HaltReason – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_LSE_MIT_HaltReason`	QuantFEED® tag name.
Numeric ID	`14752`	QuantFEED® unique ID broadcast on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`String`	String data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the reason of halting an instrument.
Possible Values	`101`	Instrument-Level Circuit Breaker Tripped
	`102`	Connectivity Issues Being Experienced by Clients
	`9998`	Matching Partition Suspended
	`9999`	System Suspended
	`Space`	Reason not available

When an instrument is no longer halted, the tag `MARKET_LSE_MIT_HaltReason` is reset by sending a value with the syntax `UNKNOWN`.

2.3.2.3. Trading Status Details

Each time a modification of the instrument status occurs, the values of the quotation tag **Trading Status Details** conveyed on the JSE MIT market data stream are disseminated via S&P Capital IQ's Real-Time Solutions's data stream in *Other Values* to indicate the current status of the instrument:

- in the callback carrying the Level1 event `notif_TradeEventExt()`, for C++
- in the event handler `TradeEventExtEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

QuantFEED® implementation of the tag `MARKET_JSE_MIT_TradingStatusDetails` is described in the table below:

Table 10 MARKET_JSE_MIT_TradingStatusDetails – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_JSE_MIT_TradingStatusDetails`	QuantFEED® tag name.
Numeric ID	`14970`	QuantFEED® unique ID disseminated on S&P Capital IQ's Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the current status of an instrument.
Possible Values	`H`	Halt
	`T`	Continuous Trading
	`a`	Opening Auction Call (Pre-Open)
	`b`	Post-Close
	`c`	Market Close (Closed)
	`d`	Closing Auction Call
	`e`	Volatility Auction Call
	`f`	Re-Opening Auction Call
	`l`	Pause
	`p`	Futures Close Out
	`s`	Closing Price Cross (Ready To Trade). **Note:** This phase immediately follows the Closing Price Publication phase (z). You can trade, but only at the closing auction price.
	`u`	Intra-Day Auction Call
	`v`	End of Trade Reporting
	`w`	No Active Session
	`x`	End of Post Close
	`y`	Pre-Trading (Start of Trading)
	`z`	Closing Price Publication (Not Available for Trading)

3. Official Closing Price

On the market JSE MIT, the closing price is provided by the market. If it is not sent by the market, the last trade is used instead. When a stock splits, the closing price is adjusted after the closing. There is no settlement price.

The market sends the closing price at the end of the **Closing Price Cross** phase (TradingStatus 17 – **Ready to Trade**), following the **Closing Price Publication** phase (TradingStatus 18 – **Not Available for Trading**), as shown in the example below:

```
TE      2013-10-21 14:47:22:114 2013-10-21 14:47:22:000 486540584       *       *       *
*       3882    8@1
TE      2013-10-21 14:47:54:176 2013-10-21 14:47:54:000 486540584       *       *       *
*       3881    8@1
TE      2013-10-21 14:49:30:353 2013-10-21 14:49:30:000 486540584       *       *       *
*       3900    10@1
VU      2013-10-21 15:50:00:160 2013-10-21 15:50:00:086.988     486540584
MARKET_JSE_MIT_TradingStatusDetails=d MARKET_LSE_MIT_HaltReason=?     TradingStatus=5
VU      2013-10-21 15:50:00:180 2013-10-21 15:50:00:098.199     486540584
LastAuctionPrice=0      LastAuctionVolume=0
VU      2013-10-21 16:00:23:108 2013-10-21 16:00:23:043.043     486540584
MARKET_JSE_MIT_TradingStatusDetails=z TradingStatus=18
VU      2013-10-21 16:15:23:098 2013-10-21 16:15:23:027.992     486540584
MARKET_JSE_MIT_TradingStatusDetails=s TradingStatus=17
TE      2013-10-21 16:15:23:099 2013-10-21 16:15:23:033.756     486540584       *       *
3812    10000@1 *       *
TE      2013-10-21 16:15:23:099 2013-10-21 16:15:23:034.242     486540584       3812    10000
2600    20@2    *       *
TE      2013-10-21 16:15:23:100 2013-10-21 16:15:23:034.750     486540584       *       *
3812    5000@1  *       *
TE      2013-10-21 17:00:23:078 2013-10-21 17:00:23:011.542     486540584       *       *
2600    20@2    *       *
SI      2013-10-21 17:00:23:078 2013-10-21 17:00:23:011.542     486540584       CLOSE   3812
TE      2013-10-21 17:00:23:078 2013-10-21 17:00:23:011.542     486540584       3812    *
*       *       *       *       C
VU      2013-10-21 17:00:23:078 2013-10-21 17:00:23:011.542     486540584
MARKET_JSE_MIT_TradingStatusDetails=b TradingStatus=18
VU      2013-10-21 17:30:23:080 2013-10-21 17:30:23:014.277     486540584
MARKET_JSE_MIT_TradingStatusDetails=x
TE      2013-10-21 17:50:01:012 2013-10-21 17:50:00:529.307     486540584       *       *
2600    10@1    *       *
TE      2013-10-21 17:50:01:012 2013-10-21 17:50:00:529.652     486540584       *       *
!       0       *       *
VU      2013-10-21 17:50:01:012 2013-10-21 17:50:00:529.652     486540584
MARKET_JSE_MIT_TradingStatusDetails=c
```

4. Finding the Latest Information

For the latest documentation and product updates, additional support and training, please contact our support services one of the following ways:

- E-mail: support@quanthouse.com
- Web: http://support.quanthouse.com.