S&P Capital IQ Real-Time Solutions

FeedOS™ Feed Description

TURQUOISE MIT

Reference n°: 20150319 – 25063 – 25923

S&P CAPITAL IQ
McGRAW HILL FINANCIAL

S&P Capital IQ Real-Time Solutions
FeedOS™ Feed Description: TURQUOISE MIT
Reference 20150319 – 25063 – 25923
March 20, 2015

France Offices
52 Rue de la Victoire
75009 Paris
France
Tel: +33 (0) 1 73 02 32 11

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

UK Office
20 Canada Square
Canary Wharf
London E14 5LH
United Kingdom
Tel: +44 (0) 203 107 1676

Singapore Office
12 Marina Boulevard
#23-01 Marina Bay
Financial Centre Tower 3
Singapore 018982
Tel: +65 6530 6546

www.capitaliq.com

TABLE OF CONTENTS

FEEDOS™ TURQUOISE MIT FEED DESCRIPTION

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

The topics this feed description covers include:

1. Referential Data

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

1.1. Available Markets and Branches

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

1.1.1. Markets

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

Table 1 List of markets available on the TURQUOISE MIT market data stream

FeedOS Market ID	Market
TRQX	Turquoise

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

```
MARKETS
market # 428    CC=GB/UNITED KINGDOM/LONDON,DESCR=TURQUOISE,WEB=www.tradeturquoise.com/
    MIC = TRQX
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
```

1.1.2. Branches

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

```
BRANCHES
    { TRQX CS   ESXXXX } qty: 2171
```

1.2. Types of Instruments

The following sections describe the instruments available on the TURQUOISE MIT market data stream, according to their type:

- 1.2.1. Equities.

1.2.1. Equities

The sample below illustrates the details of an equity:

```
instr # 428/9067 = 897590123
    PriceCurrency               string{SEK}
    Symbol                      string{QLRO}
    Description                 string{QLIRO GROUP}
    MaxFloor                    float64{447150}
    SecurityType                string{CS}
    FOSMarketId                 TRQX
    CFICode                     string{ESXXXX}
    SecuritySubType             string{EQ}
    MarketSegmentID             string{SE}
    InternalCreationDate        Timestamp{2015-01-12 02:10:00:700}
    InternalModificationDate    Timestamp{2015-01-12 02:10:00:700}
    InternalSourceId            uint16{19}
    LocalCodeStr                string{QLROs}
    ForeignFOSMarketId          XSTO
    ForeignMarketId             string{XSTO}
    ISIN                        string{SE0003652163}
    PriceIncrement_dynamic_TableId      uint32{1245306}
    UMTF                        string{QLROs}
    OperatingMIC                string{TRQX}
    MARKET_TURQUOISE_Ticker     string{QLROs}
```

1.3. Specific Referential Tags

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

- 1.3.1. MarketSegmentID
- 1.3.2. OperatingMIC
- 1.3.3. MARKET_TURQUOISE_Ticker.

1.3.1. MarketSegmentID

The values of the referential tag **MarketSegmentID** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Referential* to specify the ID of the market segment.

FeedOS implementation of the tag `MarketSegmentID` is described in the table below:

Table 2 MarketSegmentID – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MarketSegmentID`	FeedOS tag name.
Numeric ID	`1300`	FeedOS unique ID disseminated on the S&P Capital IQ Real-Time Solutions 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 ID of the market segment.

Table 2 MarketSegmentID – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	**Values Before 2013-12-02**	**Values After 2013-12-02**
	WBAH	AT
	XBRU	BE
	XSWX	CH
	XVTX	CH
	XPRA	CZ
	XETR	DE
	XCSE	DK
	XLON	EB
	XLUX	EB
	XMCE	ES
	XHEL	FI
	XAMS	FR
	XPAR	FR
	XLON	GB
	XBUD	HU
	XDUB	IE
	MTAA	IT
	XAMS	NL
	XOSL	NO
	XLIS	PT
	XSTO	SE
	ARCX	US
	XNGS	US
	XNMS	US
	XNYS	US

1.3.2. OperatingMIC

The values of the referential tag **OperatingMIC** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Referential* to specify the parent MIC.

FeedOS implementation of the tag `OperatingMIC` is described in the table below:

Table 3 OperatingMIC – technical implementation in FeedOS

Component	Value	Description
Tag Name	`OperatingMIC`	FeedOS tag name.
Numeric ID	9533	FeedOS unique ID disseminated on the S&P Capital IQ Real-Time Solutions 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	TRQX	Parent MIC for all TURQUOISE MIT branches.

1.3.3. MARKET_TURQUOISE_Ticker

The referential tag **MARKET_TURQUOISE_Ticker** is disseminated via FeedOS market data stream in *Referential* to uniquely identify the companies that are publicly traded on the market.

FeedOS implementation of the tag `MARKET_TURQUOISE_Ticker` is described in the following table:

Table 4 MARKET_TURQUOISE_Ticker – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`MARKET_TURQUOISE_Ticker`	FeedOS tag name.
Numeric ID	11300	FeedOS unique ID disseminated on the S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`String`	String data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, uniquely identifying the companies that are publicly traded on the market.

2. Quotation Data

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

- 2.1. Quotation Values
- 2.2. TradingStatus
- 2.3. Specific Quotation Tags
- 2.4. MBL, MBO and BBO Data.

2.1. Quotation Values

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

```
InstrumentStatusL1
-- 428/9067
        BID: 16.1       0       *NO ORDER*
        ASK: 16.1       0       *NO ORDER*
        LastPrice                       float64{16.2}
        LastTradeQty                    float64{592}
        DailyHighPrice                  float64{16.2}
        DailyLowPrice                   float64{16.2}
        DailyTotalVolumeTraded          float64{592}
        DailyTotalAssetTraded           float64{9590.4}
        LastTradePrice                  float64{16.2}
        LastTradeTimestamp              Timestamp{2015-03-19 08:20:12:215}
        InternalDailyOpenTimestamp      Timestamp{2015-03-19 07:59:59:032}
        InternalDailyCloseTimestamp     Timestamp{2015-03-18 16:30:14:170}
        InternalDailyHighTimestamp      Timestamp{2015-03-19 08:20:12:215}
        InternalDailyLowTimestamp       Timestamp{2015-03-19 08:20:12:215}
        InternalPriceActivityTimestamp  Timestamp{2015-03-19 10:44:25:614}
        TradingStatus                   18=NotAvailableForTrading
        LastOffBookTradePrice           float64{16.25}
        LastOffBookTradeQty             float64{50}
        LastOffBookTradeTimestamp       Timestamp{2015-03-13 14:02:07:655}
        DailyOpeningPrice               float64{16.2}
        PreviousDailyTotalVolumeTraded  float64{25645}
        PreviousDailyTotalAssetTraded   float64{405592.4}
        PreviousDailyClosingPrice       float64{16}
        PreviousBusinessDay             Timestamp{2015-03-18}
        CurrentBusinessDay              Timestamp{2015-03-19}
        DailyTotalOffBookVolumeTraded   float64{0}
        DailyTotalOffBookAssetTraded    float64{0}
        PriceActivityMarketTimestamp    Timestamp{2015-03-19 10:44:25:613}
        MARKET_TURQUOISE_DarkBookTradingStatus  Enum{5}
        MARKET_TURQUOISE_OffBookReportingTradingStatus  Enum{17}
        InternalDailyClosingPriceType   char{d}
```

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

2.2. TradingStatus

Each time a modification of the trading status occurs, the values of the quotation tag **TradingStatus** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS 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.

FeedOS implementation of the tag `TradingStatus` is described in the following table:

Table 5 TradingStatus – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`TradingStatus`	FeedOS tag name.
Numeric ID	9100	FeedOS unique ID disseminated on the S&P Capital IQ Real-Time Solutions 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
	5	Price Indication
	17	Ready to Trade
	18	Not Available for Trading

2.3. Specific Quotation Tags

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

- 2.3.1.1. AggressorSide
- 2.3.1.2. MARKET_TURQUOISE_TradeTypeIndicator.

2.3.1.1. AggressorSide

Each time a trade occurs, the values of the quotation context tag **AggressorSide** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Context*, to indicate whether the aggressor is a buyer or a seller:

- 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.

FeedOS implementation of the values currently available for the tag `AggressorSide` is described in the following table:

Table 6 AggressorSide – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`AggressorSide`	FeedOS tag name.
Numeric ID	`9356`	FeedOS unique ID disseminated on the S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating whether the aggressor is a buyer or a seller.
Possible Values	`Space`	No aggressor
	`1`	Buy Side
	`2`	Seller Side

2.3.1.2. MARKET_TURQUOISE_TradeTypeIndicator

Each time a trade occurs, the values of the quotation tag **MARKET_TURQUOISE_TradeTypeIndicator** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Context* to identify the type of trade:

- 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.

FeedOS implementation of the tag `MARKET_TURQUOISE_TradeTypeIndicator` is described in the table below:

Table 7 MARKET_TURQUOISE_TradeTypeIndicator – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_TURQUOISE_TradeTypeIndicator`	FeedOS tag name.
Numeric ID	`15300`	FeedOS unique ID broadcast on the S&P Capital IQ Real-Time Solutions 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 the type of trade.
Possible Values	`Space or Empty`	Normal Trade
	`M`	Dark Midpoint Book

2.3.2. Other Values

The following subsections describe the other values on the TURQUOISE MIT market data stream:

- 2.3.2.1. InternalDailyClosingPriceType
- 2.3.2.2. MARKET_TURQUOISE_HaltReason
- 2.3.2.3. MARKET_TURQUOISE_DarkBookTradingStatus
- 2.3.2.4. MARKET_TURQUOISE_OffBookReportingTradingStatus.

2.3.2.1. InternalDailyClosingPriceType

The values of the quotation tag **InternalDailyClosingPriceType** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the type of the internal daily closing price:

- 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.

FeedOS implementation of the tag `InternalDailyClosingPriceType` is described in the table below (the values currently disseminated are highlighted in `green`):

Table 8 InternalDailyClosingPriceType – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`InternalDailyClosingPriceType`	FeedOS tag name.
Numeric ID	`9155`	FeedOS unique ID disseminated on S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`Char`	Char data type.
Format	*`[Internal Specific Value]`*	An ***internal specific value***, detailing the type of daily closing price, as described below.
Possible Values	0	**Undefined**
	a	**Official Close** – Explicit closing price value calculated and distributed by an exchange for the main trading session of a given trading day.
	b	**Official Indicative** – Exchange has provided an indicative price and marked it as indicative, however no trading activity is observed.
	c	**Official Carry Over** – Explicit Closing price value from a previous trading day carried forward by the exchange to the given trading day.
	d	**Last Price** – Final price disseminated by the exchange for the main trading session or dissemination period of a given trading day (for indices).
	e	**Last Eligible Price** – Execution price of the final trade (subject to trade qualifiers) accepted by the exchange for the main trading session of a given trading day.
	z	**Manual** – Price disseminated manually (in case of production correction).

2.3.2.2. MARKET_TURQUOISE_HaltReason

Each time an instrument is halted from trading, the values of the quotation tag **MARKET_TURQUOISE_HaltReason** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS 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.

FeedOS implementation of the tag `MARKET_TURQUOISE_HaltReason` is described in the table below:

Table 9 MARKET_TURQUOISE_HaltReason – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_TURQUOISE_HaltReason`	FeedOS tag name.
Numeric ID	14720	FeedOS unique ID broadcast on the S&P Capital IQ Real-Time Solutions 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 for an instrument.
Possible Values	9998	Matching partition suspended
	9999	System suspended
	space	Reason not available
	1	System problem
	2	Fast market
	3	News pending

When an instrument is no longer halted, the tag `MARKET_TURQUOISE_HaltReason` is reset. To reset the tag, send a value with the syntax `UNKNOWN`.

For more details about the procedure, see the C++ code sample below:

```
FeedOS::Types::ListOfQuotationVariable const & values = inData.getValues();
for (FeedOS::Types::ListOfQuotationVariable::const_iterator it = values.begin(); it !=
values.end(); ++it) {
     unsigned int tag_num = it->getNum();
     switch(tag_num) {
          case FeedOS::TAG_MARKET_TURQUOISE_HaltReason:
          {
               Any halt_reason_value = it->getValue();
               if (halt_reason_value.get_syntax() == Syntax_UNKNOWN) {
                    // reset HaltReason
               } else {
                    // get reason
                    std::string reason_code = halt_reason_value.get_String();
               }
          }
          break;
     }
}
```

2.3.2.3. MARKET_TURQUOISE_DarkBookTradingStatus

The values of the quotation tag **MARKET_TURQUOISE_DarkBookTradingStatus** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the trading status:

- 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.

FeedOS implementation of the tag `MARKET_TURQUOISE_DarkBookTradingStatus` is described in the table below:

Table 10 MARKET_TURQUOISE_DarkBookTradingStatus – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_TURQUOISE_DarkBookTradingStatus`	FeedOS tag name.
Numeric ID	`14721`	FeedOS unique ID broadcast on the S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`Enum`	Enum data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating the trading status of the Dark Book.
Possible Values	`2`	Trading Halt
	`5`	Price Indication
	`17`	Ready to Trade
	`18`	Not Available for Trading

2.3.2.4. MARKET_TURQUOISE_OffBookReportingTradingStatus

Each time a trade occurs, the values of the quotation tag **MARKET_TURQUOISE_OffBookReportingTradingStatus** conveyed on the TURQUOISE MIT market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the off book trading status:

- 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.

FeedOS implementation of the tag `MARKET_TURQUOISE_OffBookReportingTradingStatus` is described in the table below:

Table 11 MARKET_TURQUOISE_OffBookReportingTradingStatus – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_TURQUOISE_OffBookReporting TradingStatus`	FeedOS tag name.
Numeric ID	`14722`	FeedOS unique ID broadcast on the S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`Enum`	Enum data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating the off book trading status.
Possible Values	`2`	Trading Halt
	`5`	Price Indication
	`17`	Ready to Trade
	`18`	Not Available for Trading

2.4. MBL, MBO and BBO Data*

The MBL book has a 10-level depth. The MBO book is full depth.

3. Special Behavior

A flag is set among the content mask to distinguish the Off Book Trades. For more details about this type of trades, see *FeedOS Quotation Tags Guide*.

Moreover, the Dark Book Trades are flagged as Off Book Trades and they have a specific trade type and trade condition.

4. Official Closing Price

The closing price is the last trade price upon close, as provided by the exchange. There is no correction or settlement price.

5. 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: rts-support@spcapitaliq.com
- Web: https://support.quanthouse.com.

* The MBL, MBO and BBO data may not be included by default in your Level1 data subscription, but sold separately. Depending on your contract, additional terms, conditions and fees may apply. For more details about the subscription options, please contact S&P Capital IQ Real-Time Solutions.