S&P Capital IQ Real-Time Solutions

FeedOS™ Feed Description

NGM

Reference n°: 20150512 – 25635 – 26167 – 26550

S&P Capital IQ Real-Time Solutions
FeedOS™ Feed Description: NGM
Reference 20150512 – 25635 – 26167 – 26550
May 12, 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 Offices
20 Canada Square
Canary Wharf
London E14 5LH
United Kingdom
Tel: +44 (0) 203 107 1676

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

www.spcapitaliq.com

TABLE OF CONTENTS

FEEDOS™ NGM 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 NGM 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 NGM market data stream, in terms of:

1.1. Available Markets and Branches

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

1.1.1. Markets

The NGM market data stream broadcasts informations about the following markets:

Table 1 List of markets available on the NGM market data stream

FeedOS Market ID	Market
XNGM	NGM – Nordic Growth Market
NMTF	Nordic MTF

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

```
MARKETS
market # 252    CC=SE/SWEDEN/STOCKHOLM,DESCR=NORDIC GROWTH MARKET,WEB=www.ngm.se
    MIC = XNGM
    TimeZone = Europe/Stockholm
    Country = Sweden
    NbMaxInstruments = 2000000
market # 487    CC=SE/SWEDEN/STOCKHOLM,DESCR=NORDIC MTF,WEB=www.nordicmtf.se
    MIC = NMTF
    TimeZone = Europe/Stockholm
    Country = Sweden
    NbMaxInstruments = 2000000
```

1.1.2. Branches

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

```
BRANCHES
    { XNGM CB   DCXXXX } qty: 1
    { XNGM CS   ESXXXX } qty: 13
    { XNGM GO   DBXXXX } qty: 208
    { XNGM NONE DTVSXX } qty: 210
    { XNGM NONE MMXXXX } qty: 342
    { XNGM WAR  RMXXXX } qty: 9
    { XNGM WAR  RSXXXX } qty: 11
    { XNGM WAR  RWXXXX } qty: 22723
    { NMTF CB   DCXXXX } qty: 3
    { NMTF CS   ESXXXX } qty: 21
    { NMTF WAR  RSXXXX } qty: 14
```

1.2. Types of Instruments

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

1.2.1. Equities

The sample below illustrates the details of an equity:

```
instr # 487/1092 = 1021314116
    PriceCurrency               string{SEK}
    Symbol                      string{METV MTF UR}
    Description                 string{METALLVÄRDEN I SVERIGE UR}
    SecurityType                string{CS}
    FOSMarketId                 NMTF
    PriceType                   uint8{2}
    CFICode                     string{ESXXXX}
    RoundLot                    float64{1}
    MinTradeVol                 float64{1}
    MarketSegmentID             string{MST}
    InternalCreationDate        Timestamp{2015-01-22 00:00:00:108}
    InternalModificationDate    Timestamp{2015-02-04 06:00:01:608}
    InternalHideFromLookup      bool{True}
    InternalSourceId            uint16{198}
    InternalAggregationId       uint16{198}
    InternalEntitlementId       int32{1068}
    DelayedFeedMin              uint16{10}
    LocalCodeStr                string{2952}
    ISIN                        string{SE0006731725}
    PriceIncrement_dynamic_TableId      uint32{12976229}
    OperatingMIC                string{XNGM}
    SegmentMIC                  string{NMTF}
```

1.2.2. Bonds

The sample below illustrates the details of a bond:

```
instr # 252/6573 = 528488877
    PriceCurrency               string{EUR}
    Symbol                      string{NBFC NDF 4597}
    Description                 string{Företagslåneobligation High Yield 1/2017}
    SecurityType                string{GO}
    FOSMarketId                 XNGM
    PriceType                   uint8{1}
    CFICode                     string{DBXXXX}
    RoundLot                    float64{1000}
    MinTradeVol                 float64{1000}
    MarketSegmentID             string{DFMP}
    InternalCreationDate        Timestamp{2014-03-30 02:30:02:127}
    InternalModificationDate    Timestamp{2015-03-18 06:00:00:775}
    InternalSourceId            uint16{198}
    InternalAggregationId       uint16{198}
    InternalEntitlementId       int32{1068}
    DelayedFeedMin              uint16{10}
    LocalCodeStr                string{YKY}
    ISIN                        string{SE0004328615}
    PriceIncrement_dynamic_TableId      uint32{12976228}
    OperatingMIC                string{XNGM}
```

1.2.3. Warrants

The sample below illustrates the details of a warrant:

```
instr # 252/32145 = 528514449
    PriceCurrency               string{SEK}
    Symbol                      string{B LONGTLSN P CBK}
    Description                 string{B LONGTLSN P CBK}
    SecurityType                string{WAR}
    FOSMarketId                 XNGM
    PriceType                   uint8{2}
    CFICode                     string{RWXXXX}
    RoundLot                    float64{1}
    MarketSegmentID             string{DSKO}
    InternalCreationDate        Timestamp{2014-05-12 00:00:02:307}
    InternalModificationDate    Timestamp{2015-01-08 09:12:11:247}
    InternalHideFromLookup      bool{True}
    InternalSourceId            uint16{198}
    InternalAggregationId       uint16{198}
    InternalEntitlementId       int32{1068}
    DelayedFeedMin              uint16{10}
    LocalCodeStr                string{2OAX}
    ISIN                        string{DE000CB5UK89}
    PriceIncrement_dynamic_TableId      uint32{12976228}
    OperatingMIC                string{XNGM}
```

1.2.4. Note

The sample below illustrates the details of a note:

```
instr # 252/16106 = 528498410
    PriceCurrency               string{SEK}
    Symbol                      string{RBSC 680 180328}
    Description                 string{Non-Capital Protected Notes linked to an Index}
    SecurityType                string{NONE}
    FOSMarketId                 XNGM
    PriceType                   uint8{1}
    CFICode                     string{DTVSXX}
    RoundLot                    float64{10000}
    MinTradeVol                 float64{10000}
    MarketSegmentID             string{DSSP}
    InternalCreationDate        Timestamp{2014-03-30 02:30:01:010}
    InternalModificationDate    Timestamp{2015-03-18 06:00:00:808}
    InternalSourceId            uint16{198}
    InternalAggregationId       uint16{198}
    InternalEntitlementId       int32{1068}
    DelayedFeedMin              uint16{10}
    LocalCodeStr                string{1GVB}
    ISIN                        string{SE0004976942}
    PriceIncrement_dynamic_TableId      uint32{12976228}
    OperatingMIC                string{XNGM}
```

1.3. Specific Referential Tags

The following sections describe specific referential tags available on the NGM market data stream:

- 1.3.1. OperatingMIC and SegmentMIC.

1.3.1. OperatingMIC and SegmentMIC

The values of the referential tags **OperatingMIC** and **SegmentMIC** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Referential* to specify the parent and child MIC.

FeedOS implementation of the tags `OperatingMIC` and `SegmentMIC` is described in the table below:

Table 2 OperatingMIC and SegmentMIC – technical implementation in FeedOS

Component	Value		Description
Tag Name	`OperatingMIC`	`SegmentMIC`	FeedOS tag name.
Numeric ID	`9533`	`9534`	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`	String data type.
Format	*`[Exchange Specific Value]`*	*`[Exchange Specific Value]`*	An ***exchange specific value***, specifying the parent and child MICs.
Possible Values	`XNGM`	`NMTF`	Market places of NGM.

2. Quotation Data

The following sections describe the characteristics of the quotation data on the NGM 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 the NGM market data stream:

```
InstrumentStatusL1
-- 252/36064
        BID: 70 0       *NO ORDER*
        ASK: 78 0       *NO ORDER*
        LastPrice                       float64{70}
        LastTradeQty                    float64{1}
        DailyHighPrice                  float64{70}
        DailyLowPrice                   float64{60}
        DailyTotalVolumeTraded          float64{27}
        DailyTotalAssetTraded           float64{1796}
        LastTradePrice                  float64{70}
        LastTradeTimestamp              Timestamp{2015-04-29 13:41:44:875}
        InternalDailyOpenTimestamp      Timestamp{2015-04-29 13:36:02:283}
        InternalDailyCloseTimestamp     Timestamp{2015-04-23 17:00:00:110}
        InternalDailyHighTimestamp      Timestamp{2015-04-29 13:38:57:831}
        InternalDailyLowTimestamp       Timestamp{2015-04-29 13:36:02:283}
        InternalPriceActivityTimestamp  Timestamp{2015-04-29 13:52:15:197}
        TradingStatus                   5=PriceIndication
        DailyOpeningPrice               float64{60}
        PreviousDailyTotalVolumeTraded  float64{38100}
        PreviousDailyTotalAssetTraded   float64{1383920}
        PreviousDailyClosingPrice       float64{340}
        PreviousBusinessDay             Timestamp{2015-04-23}
        CurrentBusinessDay              Timestamp{2015-04-29}
        LastAuctionPrice                float64{62}
        LastAuctionVolume               float64{11}
        LastAuctionImbalanceSide        char{B}
        LastAuctionImbalanceVolume      float64{30}
        InternalDailyClosingPriceType   char{a}
        InternalLastAuctionTimestamp    Timestamp{2015-04-29 13:39:49:478}
        PriceActivityMarketTimestamp    Timestamp{2015-04-29 13:52:15:194}
        MARKET_NGM_KnockOutBuyback      char{S}
```

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 NGM 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 3 TradingStatus – technical implementation in FeedOS

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
	15	New Price Indication
	17	Ready to Trade
	18	Not Available for Trading

2.3. Specific Quotation Tags

The following sections describe the quotation tags on the NGM 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 NGM market data stream:

- 2.3.1.1. Trade Condition.

2.3.1.1. Trade Condition

Each time a trade occurs, the values of the quotation tag **Trade Condition** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Context*:

- 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 `TradeCondition` is described in the table below:

Table 4 TradeCondition – technical implementation in FeedOS

Component	Value	Description
Tag Name	`TradeCondition`	FeedOS tag name.
Numeric ID	`277`	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 conditions of a trade.
Possible Values	`I`	Sold Last (Late Reporting)
	`AV`	Outside Spread
	`XAO`	Opening auction Trade
	`XAC`	Closing auction Trade
	`XAD`	Circuit breaker dynamic auction Trade
	`XAS`	Circuit breaker static auction Trade
	`XB`	Knock out buyback trade
	`XD`	Distribution trade
	`X0`	Outside Spread Unknown
	`XS`	Sold out buyback trade

2.3.2. Other Values

The following sections describe the other values available on the NGM market data stream:

2.3.2.1. LastAuctionPrice

The values of the quotation tag **LastAuctionPrice** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Other Values* to detail the last 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 `LastAuctionPrice` is described in the following table:

Table 5 LastAuctionPrice – technical implementation in FeedOS

Component	Value	Description
Tag Name	`LastAuctionPrice`	FeedOS tag name.
Numeric ID	9146	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	`Float64`	Float64 data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the last auction price.

2.3.2.2. LastAuctionVolume

The values of the quotation tag **LastAuctionVolume** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Other Values* to detail the last volume:

- 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 `LastAuctionVolume` is described in the following table:

Table 6 LastAuctionVolume – technical implementation in FeedOS

Component	Value	Description
Tag Name	`LastAuctionVolume`	FeedOS tag name.
Numeric ID	9147	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	`Float64`	Float64 data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the last auction volume.

2.3.2.3. LastAuctionImbalanceSide

The values of the quotation tag **LastAuctionImbalanceSide** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the imbalance side of a closing auction:

- 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 `LastAuctionImbalanceSide` is described below:

Table 7 LastAuctionImbalanceSide – technical implementation in FeedOS

Component	Value	Description
Tag Name	`LastAuctionImbalanceSide`	FeedOS tag name.
Numeric ID	`9151`	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	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the imbalance side of a closing auction.
Possible Values	`B`	Buy
	`S`	Sell

2.3.2.4. LastAuctionImbalanceVolume

The values of the quotation tag **LastAuctionImbalanceVolume** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the imbalance volume of a closing auction:

- 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 available for the tag `LastAuctionImbalanceVolume` is described below:

Table 8 LastAuctionImbalanceVolume – technical implementation in FeedOS

Component	Value	Description
Tag Name	`LastAuctionImbalanceVolume`	FeedOS tag name.
Numeric ID	`9152`	FeedOS unique ID disseminated on S&P Capital IQ Real-Time Solutions 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 imbalance volume of a closing auction.

2.3.2.5. InternalDailyClosingPriceType

The values of the quotation tag **InternalDailyClosingPriceType** conveyed on the NGM 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 9 InternalDailyClosingPriceType – technical implementation in FeedOS

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).

2.3.2.6. MARKET_NGM_KnockOutBuyback

The values of the quotation tag **MARKET_NGM_KnockOutBuyback** conveyed on the NGM market data stream are disseminated via FeedOS data stream in *Other Values* to detail the type of buyback for a knock-out product:

- 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_NGM_KnockOutBuyback` is described in the table below:

Table 10 MARKET_NGM_KnockOutBuyback – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_NGM_KnockOutBuyback`	FeedOS tag name.
Numeric ID	`15040`	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***, detailing the particular condition applicable to the trade.

Table 10 MARKET_NGM_KnockOutBuyback – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	D	Circuit breaker dynamic
	S	Circuit breaker static
	U	Sold-out buyback
	V	Distribution
	W	Knock out
	X	Knock out buyback
	Y	Knock out soft
	Z	Under observation

2.4. MBL, MBO and BBO Data*

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

3. Official Closing Price

The closing price is the last trade price upon close, as provided by the exchange. When a stock splits, the closing price is adjusted after the closing. There is no settlement price.

4. Special Behavior

The following sections describe the special behavior of the NGM market data stream:

- 4.1. Post-Open Kinematics
- 4.2. Knocked Out Instruments
- 4.3. Circuit Breaker
- 4.4. Microsecond Timestamp Precision on the Level1 Market Data.

* 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.

4.1. Post-Open Kinematics

In the Post-Open kinematics **before 2014-03-17**, during the Post-Trading Hours (usually, after 17:30 Exchange Standard Time), an instrument had the Trading Status `5=Price Indication`, as shown in the example below:

```
"TE (TradeEvent) : MARKET_TIME INSTRUMENT LAST_PRICE TRADE_QTY BID_PRICE BID_QTY ASK_PRICE
ASK_QTY *CONTENT_MASK* *FLAGS*"
"VU (ValuesUpdate) : SERVER_TIME INSTRUMENT VALUES..."
"SI (TradeEvent) *SIGNAL* : SERVER_TIME INSTRUMENT SIGNAL LAST_PRICE"

TE  16:24:51:065 528493787       *       *       417.63  5000@1  419.3   5000@1
VU  16:25:00:254 528493787       TradingStatus=5
SI  17:00:00:725 528493787       CLOSE   418.33
TE  17:00:00:725 528493787       418.33  *       *       *       *       *       C
VU  17:00:00:725 528493787       TradingStatus=18
VU  17:00:00:725 528493787       SessionVWAPPrice=3088818.98
```

In the Post-Open kinematics **after 2014-03-17**, during the same Post-Trading Hours, an instrument has the Trading Status `15=New Price Indication`, as shown in the example below:

```
"TE (TradeEvent) : MARKET_TIME INSTRUMENT LAST_PRICE TRADE_QTY BID_PRICE BID_QTY ASK_PRICE
ASK_QTY *CONTENT_MASK* *FLAGS*"
"VU (ValuesUpdate) : SERVER_TIME INSTRUMENT VALUES..."
"SI (TradeEvent) *SIGNAL* : SERVER_TIME INSTRUMENT SIGNAL LAST_PRICE"

TE  16:24:51:065 528493787       *       *       417.63  5000@1  419.3   5000@1
VU  16:25:00:254 528493787       TradingStatus=15
SI  17:00:00:725 528493787       CLOSE   418.33
TE  17:00:00:725 528493787       418.33  *       *       *       *       *       C
VU  17:00:00:725 528493787       TradingStatus=18
VU  17:00:00:725 528493787       SessionVWAPPrice=3088818.98
```

Moreover, in the Post-Trading Hours, the session orders and the order changes are still not disclosed (not disseminated in the market data), although you can submit, modify and/or cancel other orders.

Caution	Effective 2014-03-17, the Trading Status value `5=Price Indication` has been completely removed from the list of possible Trading Status values on the NGM market data stream.

4.2. Knocked Out Instruments

In the Level1 Market Data Kinematics **before 2015-04-13**, when the tag `MARKET_NGM_KnockOutBuyback` received the value `Y=Knock out soft`, a halted knocked out instrument changed its Trading Status from `2=TradingHalt` to `17=ReadyToTrade` after the OPEN signal, as shown in the example below:

```
“TE (TradeEvent) : MARKET_TIME INSTRUMENT LAST_PRICE TRADE_QTY BID_PRICE BID_QTY ASK_PRICE
ASK_QTY *CONTENT_MASK* *FLAGS*"
"VU (ValuesUpdate) : SERVER_TIME INSTRUMENT VALUES..."
"SI (TradeEvent) *SIGNAL* : SERVER_TIME INSTRUMENT SIGNAL LAST_PRICE"

VU   06:59:59:682   252/28794   TradingStatus=21
VU   07:00:00:475   252/28794   MARKET_NGM_KnockOutBuyback=Y TradingStatus=2
SI   08:00:00:244   252/28794   OPEN  *
TE   08:00:00:244   252/28794   *     *     *     *     *     *     O
VU   08:00:00:244   252/28794   MARKET_NGM_KnockOutBuyback=Y TradingStatus=17
```

In the Level1 Market Data Kinematics **after 2015-04-13**, when the tag `MARKET_NGM_KnockOutBuyback` receives the value `Y=Knock out soft`, a halted knocked out instrument remains halted (Trading Status `2=TradingHalt`) even after the OPEN signal, as shown in the example below:

```
“TE (TradeEvent) : MARKET_TIME INSTRUMENT LAST_PRICE TRADE_QTY BID_PRICE BID_QTY ASK_PRICE
ASK_QTY *CONTENT_MASK* *FLAGS*"
"VU (ValuesUpdate) : SERVER_TIME INSTRUMENT VALUES..."
"SI (TradeEvent) *SIGNAL* : SERVER_TIME INSTRUMENT SIGNAL LAST_PRICE"

VU   07:00:00:475.254   252/28794   MARKET_NGM_KnockOutBuyback=Y TradingStatus=2
SI   08:00:00:244.178   252/28794   OPEN  *
TE   08:00:00:244.178   252/28794   *     *     *     *     *     *     O
VU   08:00:00:244.178   252/28794   MARKET_NGM_KnockOutBuyback=Y
```

4.3. Circuit Breaker

Effective 2015-06-01, the Circuit Breakers used to halt the trading each time an extraordinary market volatility occurs is flagged in the Level1 Market Data by the tag `MARKET_NGM_KnockOutBuyback` (see section 2.3.2.6. MARKET_NGM_KnockOutBuyback). Moreover, when the tag `MARKET_NGM_KnockOutBuyback` disseminates the value `D=CircuitBreakerDynamic` or `S=CircuitBreakerStatic`, the TradingStatus of the instrument changes to `5=PriceIndication`, as shown in the example below:

```
VU   04:00:00:100.974   528512640    TradingStatus=21
VU   04:45:00:143.912   528512640    TradingStatus=5
SI   05:00:00:103.317   528512640    OPEN    *
TE   05:00:00:103.317   528512640    *       *       *       *       *       *       O
VU   05:00:00:103.317   528512640    TradingStatus=17
VU   08:03:14:268.636   528512640    MARKET_NGM_KnockOutBuyback=D    TradingStatus=5
VU   08:03:24:369.549   528512640    TradingStatus=17        MARKET_NGM_KnockOutBuyback=?
TE   08:05:17:918.076   528512640    *       *       4       45@1    !       0
TE   08:05:33:287.851   528512640    *       *       *       *       33      100@1
TE   08:06:00:024.556   528512640    *       *       32      33@1    *       *
VU   08:06:22:611.710   528512640    MARKET_NGM_KnockOutBuyback=S    TradingStatus=5
TE   08:06:22:611.710   528512640    *       *       33      33@1    *       *
VU   08:06:22:612.046   528512640    LastAuctionPrice=33     LastAuctionVolume=33
LastAuctionImbalanceVolume=?    LastAuctionImbalanceSide=?
TE   08:06:43:852.267   528512640    *       *       33      120@1   *       *
VU   08:06:43:852.267   528512640    LastAuctionVolume=100   LastAuctionImbalanceVolume=20
LastAuctionImbalanceSide=B
TE   08:07:26:108.058   528512640    *       *       *       *       33.01   100@1
```

4.4. Microsecond Timestamp Precision on the Level1 Market Data

Effective 2015-04-13, the server timestamps display microsecond units on the Level1 Market Data, as shown in the example below (highlighted in `green`):

```
"TE (TradeEvent) : MARKET_TIME INSTRUMENT LAST_PRICE TRADE_QTY BID_PRICE BID_QTY ASK_PRICE
ASK_QTY *CONTENT_MASK* *FLAGS*"

TE    14:06:20:564.560    528504599    *    *    133    1800@9    *    *
TE    14:06:20:564.560    528504599    *    *    133    1600@8    *    *
TE    14:06:20:564.560    528504599    *    *    133    1400@7    *    *
TE    14:06:20:564.560    528504599    *    *    133    1200@6    *    *
TE    14:06:20:564.560    528504599    *    *    133    1000@5    *    *
TE    14:06:20:564.560    528504599    *    *    133    800@4     *    *
```

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.