S&P Capital IQ Real-Time Solutions

# FeedOS™ Developer's Notice

## NGM – Feed Update

Reference n°: 20150427 – 25635 – 26167 – 26397

**Effective as of: 01 June 2015***

**Action required from users: MANDATORY ACTION**

S&P CAPITAL IQ
McGRAW HILL FINANCIAL

* For the actual day when the changes to your custom feed handler take effect, please contact your QuantFEED® project manager.

S&P Capital IQ Real-Time Solutions
FeedOS™ Developer's Notice: NGM – Feed Update
Reference 20150427 – 25635 – 26167 – 26397
May 07, 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

# UPDATE OF THE NGM MARKET DATA STREAM

To reflect the changes caused by the introduction of the Order Protection Feature on the NGM Exchange, S&P Capital IQ Real-Time Solutions has decided to enhance the content of FeedOS™.

This developer's notice contains late-breaking information about the implementation of this modification in your applications, which may not be included otherwise in the published documentation. The topics this notice covers include:



# 1. Update Summary

**Table 1 Current update summary**

| | |
|---|---|
| **Notice Reference** | 20150427 – 25635 – 26167 – 26397 |
| **Exchanges** | NGM |
| **Concerned MICs** | XNGM, NMTF |
| **Internal Source ID** | 198 |
| **Effective Date** | **2015-06-01*** |
| **Impact** | • Update of the Quotation Tags<br>• Update of the Quotation Context Tags<br>• Changes to the Level1 Market Data Kinematics – Circuit Breaker |
| **Action required** | **MANDATORY ACTION** - see sections:<br>• 2.1.5. TradingStatus<br>• 2.3. Changes to the Level1 Market Data Kinematics – Circuit Breaker. |

# 2. FeedOS Technical Implementation

Effective Monday, **June 01*** **2015,** S&P Capital IQ Real-Time Solutions enhances the quotation and quotation context data, and changes the Level1 Market Data Kinematics to accommodate the new information disseminated on the NGM market data stream, as described below:



## 2.1. Changes to the Quotation Data

S&P Capital IQ Real-Time Solutions **introduces** the quotation tags below to accommodate the information disseminated on the NGM market data stream:

**Table 2 Quotation tags added on the NGM market data stream**

| Tag Name | Numeric ID | Type |
|---|---|---|
| LastAuctionPrice | 9146 | Float64 |
| LastAuctionVolume | 9147 | Float64 |
| LastAuctionImbalanceSide | 9151 | Char |
| LastAuctionImbalanceVolume | 9152 | Float64 |

Moreover, S&P Capital IQ Real-Time Solutions **updates** the quotation tags below:

**Table 3 Quotation tags disseminating updated values on the NGM market data stream**

| Tag Name | Numeric ID | Type |
|---|---|---|
| TradingStatus | 9100 | Enum |
| MARKET_NGM_KnockOutBuyback | 15040 | Char |

### 2.1.1. 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#

* This is the proposed day for the update of the standard version of the feed handler. For dedicated feed handlers, the date and Source IDs may differ. For the actual day when the changes to your custom feed handler will take effect, please contact your FeedOS™ project manager.

- in the callback carrying the Level1 event `quotNotifTradeEventExt`, for Java.

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

**Table 4 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. |

## 2.1.2. 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 5 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. |

## 2.1.3. 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 6 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 |

## 2.1.4. 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 7 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. |

## 2.1.5. 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 (newly added values are in green):

**Table 8 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 |

## 2.1.6. 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 (newly added values are in green):

**Table 9 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. |
| **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 |

### Quotation Data Sample

Below is an example showing the current implementation of the newly added (in green) and updated (in blue) quotation tags:

```
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}
```

## 2.2. Changes to the Quotation Context Data

S&P Capital IQ Real-Time Solutions **updates** the quotation tags below:

**Table 10 Quotation tags disseminating updated values on the NGM market data stream**

| Tag Name | Numeric ID | Type |
|---|---|---|
| Trade Condition | 277 | String |

### 2.2.1. 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 (newly added values are in `green`):

**Table 11 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 |

## Quotation Context Data Sample

Below is an example showing the current implementation of the updated (in `blue`) quotation tags:

```
“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   13:09:40:960.988    500     1000    !       0       !       0       HL
TradeCondition=XAS, Buyer=FID,Seller=FID
VU   13:09:40:960.988    DailyOpeningPrice=500
TE   13:20:03:464.021    *       *       200     1000@1  *       *
VU   13:20:24:825.393    TradingStatus=2
TE   13:20:24:825.393    *       *       *       *       200     1000@1
TE   13:21:03:181.476    *       *       250     1000@1  *       *
VU   13:21:03:181.827    LastAuctionPrice=356.01 LastAuctionVolume=1000
LastAuctionImbalanceVolume=?    LastAuctionImbalanceSide=?
```

## 2.3. Changes to the Level1 Market Data Kinematics – Circuit Breaker

Effective 2015-06-01, the Circuit Breakers used to halt the trading each time an extraordinary market volatility occurs will be flagged in the Level1 Market Data by the tag `MARKET_NGM_KnockOutBuyback` (see section 2.1.6. MARKET_NGM_KnockOutBuyback). Moreover, when the tag `MARKET_NGM_KnockOutBuyback` disseminates the value `D=CircuitBreakerDynamic` or `S=CircuitBreakerStatic`, the TradingStatus of the instrument will change 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
```

# 3. Finding the Latest Information

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

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