

**S&P Capital IQ Real-Time Solutions**

## **FeedOS™ Feed Description**

---

### **NASDAQ BASIC**

Reference n°: 20141015 – 22213 – 23260



S&P Capital IQ Real-Time Solutions  
FeedOS™ Feed Description: TEL AVIV  
Reference 20141015 – 22213 – 23260  
October 22, 2014

**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](http://www.capitaliq.com)

Copyright © 2014 by Standard & Poor's Financial Services LLC, a part of McGraw Hill Financial.

All rights reserved. S&P CAPITAL IQ is a trademark of Standard & Poor's Financial Services LLC. STANDARD & POOR'S, S&P, GLOBAL CREDIT PORTAL and RATINGSDIRECT are registered trademarks of Standard & Poor's Financial Services LLC.

No content (including ratings, credit-related analyses and data, valuations, model, software or other application or output therefrom) or any part thereof (Content) may be modified, reverse engineered, reproduced or distributed in any form by any means, or stored in a database or retrieval system, without the prior written permission of Standard & Poor's Financial Services LLC or its affiliates (collectively, S&P). The Content shall not be used for any unlawful or unauthorized purposes. S&P and any third-party providers, as well as their directors, officers, shareholders, employees or agents (collectively S&P Parties) do not guarantee the accuracy, completeness, timeliness or availability of the Content. S&P Parties are not responsible for any errors or omissions (negligent or otherwise), regardless of the cause, for the results obtained from the use of the Content, or for the security or maintenance of any data input by the user. The Content is provided on an "as is" basis. S&P PARTIES DISCLAIM ANY AND ALL EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR USE, FREEDOM FROM BUGS, SOFTWARE ERRORS OR DEFECTS, THAT THE CONTENT'S FUNCTIONING WILL BE UNINTERRUPTED OR THAT THE CONTENT WILL OPERATE WITH ANY SOFTWARE OR HARDWARE CONFIGURATION. In no event shall S&P Parties be liable to any party for any direct, indirect, incidental, exemplary, compensatory, punitive, special or consequential damages, costs, expenses, legal fees, or losses (including, without limitation, lost income or lost profits and opportunity costs or losses caused by negligence) in connection with any use of the Content even if advised of the possibility of such damages.

Credit-related and other analyses, including ratings, and statements in the Content are statements of opinion as of the date they are expressed and not statements of fact or recommendations to purchase, hold, or sell any securities or to make any investment decisions. S&P assumes no obligation to update the Content following publication in any form or format. The Content should not be relied on and is not a substitute for the skill, judgment and experience of the user, its management, employees, advisors and/or clients when making investment and other business decisions. S&P's opinions and analyses do not address the suitability of any security. S&P does not act as a fiduciary or an investment advisor except where registered as such. While S&P has obtained information from sources it believes to be reliable, S&P does not perform an audit and undertakes no duty of due diligence or independent verification of any information it receives.

S&P keeps certain activities of its business units separate from each other in order to preserve the independence and objectivity of their respective activities. As a result, certain business units of S&P may have information that is not available to other S&P business units. S&P has established policies and procedures to maintain the confidentiality of certain non-public information received in connection with each analytical process.

# TABLE OF CONTENTS

<b>FeedOS™ NASDAQ BASIC Feed Description</b>	<b>1</b>
1. Referential Data	1
1.1. Available Markets and Branches	1
1.1.1. Markets	1
1.1.2. Branches	2
1.2. Types of Instruments	2
1.2.1. Equities	3
1.2.2. Bonds	3
1.2.3. Rights	4
1.2.4. Warrants	4
1.3. Referential Tags	5
1.3.1. OperatingMIC	5
1.3.2. SegmentMIC	5
2. Quotation Data	6
2.1. Quotation Values	6
2.2. Trading Status	6
2.3. Specific Quotation Tags	7
2.3.1. Trade Conditions	7
2.3.1.1. TradeID	7
2.3.1.2. TradeImpactIndicator	8
2.3.1.3. MARKET_NASDAQ_UTP_SaleCondition	10
2.3.2. Other Values	11
2.3.2.1. TradingSessionId	11
2.3.2.2. RegSHOAction	11
2.3.2.3. InternalDailyClosingPriceType	12
2.3.2.4. PreviousInternalDailyClosingPriceType	13
3. Official Closing Price	14
4. Special Behavior – Extended Trading Hours	14
5. Finding the Latest Information	15



# FEEDOS™ NASDAQ BASIC 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 NASDAQ BASIC market data stream, their possible values and current FeedOS technical implementation.

The topics this feed description covers include:

- [1. Referential Data](#)
- [2. Quotation Data](#)
- [3. Official Closing Price](#)
- [4. Special Behavior – Extended Trading Hours](#)
- [5. Finding the Latest Information.](#)

## 1. Referential Data

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

- [1.1. Available Markets and Branches](#)
- [1.2. Types of Instruments](#)
- [1.3. Referential Tags.](#)

### 1.1. Available Markets and Branches

This section details the list of [Markets](#) and [Branches](#) available on the NASDAQ BASIC market data stream.

#### 1.1.1. Markets

The NASDAQ BASIC market data stream broadcasts informations about the following markets:

**Table 1** List of markets available on the NASDAQ BASIC market data stream

FeedOS Market ID	Market
FINN	FINRA / NASDAQ Trade Reporting Facility
XNAS	NASDAQ

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

```
MARKETS
market # 98      CC=US/UNITED STATES OF AMERICA/WASHINGTON,DESCR=FINRA/NASDAQ TRF (TRADE
REPORTING FACILITY),WEB=www.finra.org,OLD=EUWX,SEQNUM=1
  MIC = FINN
  TimeZone = America/New_York
  Country = US
  NbMaxInstruments = 2000000
market # 330     CC=US/UNITED STATES OF AMERICA/NEW YORK,DESCR=NASDAQ,WEB=www.nasdaq.com
  MIC = XNAS
  TimeZone = America/New_York
  Country = US
  NbMaxInstruments = 2000000
```

### 1.1.2. Branches

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

```
BRANCHES
{ FINN CS   ESXXXX } qty: 343
{ FINN CS   ESXXXX } qty: 7021
{ FINN CS   EXXXXX } qty: 201
{ FINN GO   DBXXXX } qty: 1
{ FINN MF   EUXXXX } qty: 76
{ FINN NONE EUXXXX } qty: 7
{ FINN NONE EXXXXX } qty: 24
{ FINN NONE RXXXXX } qty: 8
{ FINN NONE XXXXXX } qty: 5
{ FINN PS   EPXXXX } qty: 548
{ FINN WAR  RWXXXX } qty: 104
{ XNAS CS   ESXXXX } qty: 343
{ XNAS CS   ESXXXX } qty: 7252
{ XNAS CS   EXXXXX } qty: 201
{ XNAS GO   DBXXXX } qty: 1
{ XNAS MF   EUXXXX } qty: 76
{ XNAS NONE EUXXXX } qty: 17
{ XNAS NONE EXXXXX } qty: 24
{ XNAS NONE RXXXXX } qty: 11
{ XNAS NONE XXXXXX } qty: 28
{ XNAS PS   EPXXXX } qty: 560
{ XNAS WAR  RWXXXX } qty: 114
```

## 1.2. Types of Instruments

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

- [1.2.1. Equities](#)

- [1.2.2. Bonds](#)
- [1.2.3. Rights](#)
- [1.2.4. Warrants.](#)

### 1.2.1. Equities

The sample below illustrates the details of an equity:

```
instr # 330/3701 = 692063861
  PriceCurrency      string{USD}
  Symbol             string{SPAR}
  Description         string{Spartan Motors, Inc. - Common Stock}
  SecurityType       string{CS}
  FOSMarketId        XNAS
  CFICode            string{ESXXXX}
  RoundLot           float64{100}
  InternalCreationDate Timestamp{2014-07-26 14:44:29:398}
  InternalModificationDate Timestamp{2014-10-14 10:30:05:392}
  InternalSourceId    uint16{226}
  InternalAggregationId uint16{226}
  InternalEntitlementId int32{1143}
  LocalCodeStr        string{SPAR}
  PriceIncrement_dynamic_TableId uint32{14811236}
  MBLLayersDesc        string{0}
  OperatingMIC         string{XNAS}
  SegmentMIC           string{XNGS}
```

### 1.2.2. Bonds

The sample below illustrates the details of a bond:

```
instr # 330/47838 = 692107998
  PriceCurrency      string{USD}
  Symbol             string{EMZ}
  Description         string{Entergy Mississippi, Inc. First Mortgage Bonds, 6.0%
Series due May 1, 2051}
  SecurityType       string{GO}
  FOSMarketId        XNAS
  CFICode            string{DBXXXX}
  RoundLot           float64{100}
  InternalCreationDate Timestamp{2014-07-26 14:44:33:476}
  InternalModificationDate Timestamp{2014-10-14 10:30:05:564}
  InternalSourceId    uint16{226}
  InternalAggregationId uint16{226}
  InternalEntitlementId int32{1065}
  LocalCodeStr        string{EMZ}
  ForeignFOSMarketId  XNYS
  ForeignMarketId     string{XNYS}
  PriceIncrement_dynamic_TableId uint32{14811236}
  OperatingMIC         string{XNAS}
```

### 1.2.3. Rights

The sample below illustrates the details of a right:

```
instr # 330/106732 = 692166892
  PriceCurrency      string{USD}
  Symbol             string{BIOFR}
  Description         string{BioFuel Energy Corp. - Subscription Rights}
  SecurityType       string{NONE}
  FOSMarketId        XNAS
  CFICode            string{RXXXXX}
  RoundLot           float64{100}
  InternalCreationDate Timestamp{2014-09-23 07:00:03:658}
  InternalModificationDate Timestamp{2014-10-14 10:30:05:566}
  InternalSourceId    uint16{226}
  InternalAggregationId uint16{226}
  InternalEntitlementId int32{1143}
  LocalCodeStr        string{BIOFR}
  PriceIncrement_dynamic_TableId uint32{14811236}
  OperatingMIC         string{XNAS}
  SegmentMIC           string{XNCM}
```

### 1.2.4. Warrants

The sample below illustrates the details of a warrant:

```
instr # 98/2632 = 205523528
  PriceCurrency      string{USD}
  Symbol             string{MEILZ}
  Description         string{Methes Energies International Ltd - Class B Warrants}
  SecurityType       string{WAR}
  FOSMarketId        FINN
  CFICode            string{RWXXXX}
  RoundLot           float64{100}
  InternalCreationDate Timestamp{2014-08-04 07:00:04:188}
  InternalModificationDate Timestamp{2014-10-14 10:30:05:033}
  InternalSourceId    uint16{226}
  InternalAggregationId uint16{226}
  InternalEntitlementId int32{1143}
  LocalCodeStr        string{MEILZ}
  PriceIncrement_dynamic_TableId uint32{14811236}
  OperatingMIC         string{FINR}
  SegmentMIC           string{FINN}
```

## 1.3. Referential Tags

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

- [1.3.1. OperatingMIC](#)
- [1.3.2. SegmentMIC](#)

### 1.3.1. OperatingMIC

The values of the referential tag **OperatingMIC** conveyed on the NASDAQ BASIC 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 2      OperatingMIC – technical implementation in FeedOS**

Component	Value	Description
<b>Tag Name</b>	OperatingMIC	FeedOS tag name.
<b>Numeric ID</b>	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.
<b>Type</b>	String	String data type.
<b>Format</b>	<i>[Exchange specific value]</i>	An <i>exchange specific value</i> , specifying the parent MIC.
<b>Possible Values</b>	FINR	FINRA
	XNAS	NASDAQ – All Markets

### 1.3.2. SegmentMIC

The values of the referential tag **SegmentMIC** conveyed on the NASDAQ BASIC market data stream are disseminated via FeedOS data stream in *Referential* to specify the child MIC.

FeedOS implementation of the tag **SegmentMIC** is described in the table below:

**Table 3      SegmentMIC – technical implementation in FeedOS**

Component	Value	Description
<b>Tag Name</b>	SegmentMIC	FeedOS tag name.
<b>Numeric ID</b>	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.
<b>Type</b>	String	String data type.
<b>Format</b>	<i>[Exchange specific value]</i>	An <i>exchange specific value</i> , specifying the child MIC.
<b>Possible Values</b>	FINN	FINRA/NASDAQ TRF (Trade Reporting Facility)
	XNAS	NASDAQ – All Markets
	XNCM	NASDAQ Capital Market
	XNGS	NASDAQ/NGS (Global Select Market)
	XNMS	NASDAQ/NMS (Global Market)



## 2. Quotation Data

The sections below describe the characteristics of the quotation data on the NASDAQ BASIC market data stream, in terms of:

- [2.1. Quotation Values](#)
- [2.2. Trading Status](#)
- [2.3. Specific Quotation Tags](#).

### 2.1. Quotation Values

The example below shows the possible quotation values of an instrument on the NASDAQ BASIC market data stream:

```
InstrumentStatusL1
-- 98/2632
    BID: 0 0      *NO ORDER*
    ASK: 0 0      *NO ORDER*
    DailyTotalVolumeTraded      float64{0}
    DailyTotalAssetTraded       float64{0}
    InternalDailyOpenTimestamp   Timestamp{2014-10-14 13:30:00}
    InternalDailyCloseTimestamp  Timestamp{2014-10-10 20:00:00}
    InternalDailyHighTimestamp   Timestamp{2014-10-07 17:38:30:202}
    InternalDailyLowTimestamp    Timestamp{2014-10-07 17:38:30:202}
    InternalPriceActivityTimestamp Timestamp{2014-10-07 17:38:30:202}
    TradingStatus                17=ReadyToTrade
    RegSHOAction                 1=NoPriceTest
    PriorSessionsTotalAssetTraded float64{0}
    PriorSessionsTotalVolumeTraded float64{0}
    PriorSessionsTotalOffBookAssetTraded float64{0}
    PriorSessionsTotalOffBookVolumeTraded float64{0}
    PreviousDailyTotalVolumeTraded float64{0}
    PreviousDailyTotalAssetTraded float64{0}
    PreviousDailyClosingPrice     float64{0.198}
    PreviousBusinessDay           Timestamp{2014-09-24}
    CurrentBusinessDay            Timestamp{2014-10-14}
    PreviousInternalDailyClosingPriceType char{e}
    PriceActivityMarketTimestamp  Timestamp{2014-10-07 17:38:30:139}
    InternalDailyBusinessDayTimestamp Timestamp{2014-10-14 07:05:00:303}
```

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

### 2.2. Trading Status

Each time a modification of the trading status occurs, the values of the quotation tag **Trading Status** in the NASDAQ BASIC 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 **Trading Status** is described in the table below:

**Table 4 Trading Status of the NASDAQ BASIC market data stream – technical implementation in FeedOS**

Component	Value	Description
Tag Name	TradingStatus	FeedOS tag name.
Numeric ID	9100	FeedOS unique ID broadcast on S&P Capital IQ Real-Time Solutions data stream. It is the numeric equivalent of the tag name.
Type	Enum	Enumeration data type.
Format	<i>[Exchange Specific Value]</i>	An <b>exchange specific value</b> , as described below, concerning the characteristics of the trading status.
Possible Values	2	Trading Halt
	5	Price Indication
	17	Ready to Trade
	18	Not Available for Trading
	20	Unknown or Invalid

## 2.3. Specific Quotation Tags

The following section describe the specific quotation tags available on the NASDAQ BASIC market data stream:

- [2.3.1. Trade Conditions](#)
- [2.3.2. Other Values.](#)

### 2.3.1. Trade Conditions

The sections below describe the trade conditions on the NASDAQ BASIC market data stream:

- [2.3.1.1. TradeID](#)
- [2.3.1.2. TradeImpactIndicator](#)
- [2.3.1.3. MARKET\\_NASDAQ\\_UTP\\_SaleCondition.](#)

#### 2.3.1.1. TradeID

Each time a trade occurs, the values of the quotation context tag **TradeID** conveyed on the NASDAQ BASIC market data stream are disseminated via FeedOS data stream in *Context* to detail the unique ID assigned to the trade entity once it is received or matched by the exchange or central counterparty:

- 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 TradeID is described in the table below:

**Table 5 TradeID – technical implementation in FeedOS**

Component	Value	Description
Tag Name	TradeID	FeedOS tag name.
Numeric ID	1003	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	<i>[Exchange Specific Value]</i>	An <b>exchange specific value</b> , detailing the unique ID assigned to the trade entity once it is received or matched by the exchange or central counterparty.

### 2.3.1.2. TradeImpactIndicator

Each time a trade occurs, the values of the quotation tag **TradeImpactIndicator** conveyed on the NASDAQ BASIC market data stream are disseminated via FeedOS data stream in *Context* to specify the impact of a trade on pricing and volume data:

- 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 TradeImpactIndicator is described in the table below:

**Table 6 TradeImpactIndicator – technical implementation in FeedOS**

Component	Value	Description
Tag Name	TradeImpactIndicator	FeedOS tag name.
Numeric ID	9902	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	UInt32 Bitmask	UInt32 Bitmask data type.
Format	<i>[Exchange and Internal Specific Value]</i>	An <b>exchange and internal specific value</b> , detailing the impact of a trade on pricing and volume data.

Table 6 TradeImpactIndicator – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	Market Eligibility Rules  Static information about what type of eligibility rules an exchange is using.	TII_HasOpen  <b>Note:</b> 'TII_' is a generic abbreviation of TradeImpactIndicator. For the complete name and bit position in each API version, see the corresponding FeedOS API documentation.  The exchange uses opening price eligibility rules – Yes/No
		TII_HasHighLow  The exchange uses high/low price eligibility rules – Yes/No
		TII_HasLast  The exchange uses last price eligibility rules – Yes/No
		TII_HasVolume  The exchange uses traded volume eligibility rules – Yes/No
		TII_HasOffBookVolume  The exchange uses offbook traded volume eligibility rules – Yes/No
		TII_HasVWAP  The exchange uses volume-weighted average price eligibility rules – Yes/No
	Trade Eligibility Rules  Dynamic information (based on the exchange-specific Trade Conditions) about the impact of each trade on the daily statistics.	TII_Open  The trade price is eligible as opening price – Yes/No  <b>Note:</b> The Content Mask and/or Other Values indicate if the trade price is an Open/Last, High/Low price. For more details about price eligibility, see also <i>FeedOS Quotation Tags Guide</i> .
		TII_HighLow  The trade price is eligible as high/low price – Yes/No  <b>Note:</b> The Content Mask and/or Other Values indicate if the trade price is an Open/Last, High/Low price. For more details about price eligibility, see also <i>FeedOS Quotation Tags Guide</i> .
		TII_Last  The trade price is eligible as last price – Yes/No  <b>Note:</b> The Content Mask and/or Other Values indicate if the trade price is an Open/Last, High/Low price. For more details about price eligibility, see also <i>FeedOS Quotation Tags Guide</i> .
		TII_Volume  The trade could alter the traded volume – Yes/No
		TII_OffBookVolume  The trade could alter the offbook traded volume – Yes/No
		TII_VWAP  The trade could alter the volume-weighted average price – Yes/No

### 2.3.1.3. MARKET\_NASDAQ\_UTP\_SaleCondition

Each time an UTP participant enters a transaction, the values of the quotation tag **Sale Condition** conveyed on the NASDAQ BASIC market data stream are disseminated via S&P Capital IQ Real-Time Solutions's data stream in *Context* to indicate 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_NASDAQ_UTP_SaleCondition` is described in the table below:

**Table 7 MARKET\_NASDAQ\_UTP\_SaleCondition – technical implementation in FeedOS**

Component	Value	Description
Tag Name	MARKET_NASDAQ_UTP_SaleCondition	FeedOS tag name.
Numeric ID	15650	FeedOS unique ID broadcast on S&P Capital IQ Real-Time Solutions's data stream. This is the numeric equivalent of the tag name.
Type	String	String data type.
Format	<i>[Exchange specific value]</i>	An <b>exchange specific value</b> , indicating the type of trade transaction entered by an UTP participant.
Possible Values	<b>Level 1</b>	
	@	Regular Settlement
	C	Cash Settlement
	N	Next Day Settlement
	R	Seller Settlement
	<b>Level 2</b>	
	F	Intermarket Sweep
	O	Opening Print
	4	Derivative Priced
	5	Re-Opening Print
	6	Closing Print
	<Empty>	Not applicable
	<b>Level 3</b>	
	T	Extended Hours Trade
	U	Extended Hours Trade – Reported Late or Out of Sequence
	L	Derivative Priced
	Z	Sold Last – Reported Late But In Sequence
	<Empty>	Not applicable
	<b>Level 4</b>	
	A	Acquisition
	B	Bunched
	D	Distribution
	H	Price Variation Trade
	M	NASDAQ Official Close Price(NOCP)
	O	Odd lot execution

Table 7 MARKET\_NASDAQ\_UTP\_SaleCondition – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	P	Prior Reference Price
	Q	NASDAQ Official Opening Price (NOOP)
	T	Extended Hours Trade
	S	Split Trade
	W	Average Price
	X	Cross Trade
	x	Odd lot Cross Trade
	<Empty>	Not applicable

### 2.3.2. Other Values

The following sections describe the specific quotation tags available on the NASDAQ BASIC market data stream:

- [2.3.2.1. TradingSessionId](#)
- [2.3.2.2. RegSHOAction](#)
- [2.3.2.3. InternalDailyClosingPriceType](#)
- [2.3.2.4. PreviousInternalDailyClosingPriceType](#)

#### 2.3.2.1. TradingSessionId

The values of the quotation tag **TradingSessionId** conveyed on the NASDAQ BASIC market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the ID of the current trading session:

- 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 **TradingSessionId** is described in the following table:

Table 8 TradingSessionId – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	TradingSessionId	FeedOS tag name.
Numeric ID	9101	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	Int8	Int8 data type.
Format	<i>[Exchange Specific Value]</i>	An <b>exchange specific value</b> , indicating the ID of the current trading session.
Possible Values	-2	Late Extended Trading Hours
	-1	Early Extended Trading Hours
	1	Regular Trading Hours

#### 2.3.2.2. RegSHOAction

Each time a short sale price restriction occurs, the values of the quotation tag **Short Sale Price Restriction** conveyed on the NASDAQ BASIC 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 `RegSHOAction` is described in the table below:

**Table 9** `RegSHOAction` – technical implementation in FeedOS

Component	Value	Description
Tag Name	<code>RegSHOAction</code>	FeedOS tag name.
Numeric ID	9113	FeedOS unique ID broadcast on S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	Enum	Enum data type.
Format	<i>[Exchange Specific Value]</i>	An <b>exchange specific value</b> , detailing the short sale restriction status.
Possible Values	1	Short sale restriction deactivated – No Price Test.
	2	Short sale restriction activated – Price Test in effect.
	3	Short sale restriction continued – Price Test remains in effect.

### 2.3.2.3. InternalDailyClosingPriceType

The values of the quotation tag `InternalDailyClosingPriceType` conveyed on the NASDAQ BASIC 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 10** `InternalDailyClosingPriceType` – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	<code>InternalDailyClosingPriceType</code>	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	<i>[Internal Specific Value]</i>	An <b>internal specific value</b> , detailing the type of daily closing price, as described below.

**Table 10 InternalDailyClosingPriceType – technical implementation in QuantFEED® (Continued)**

Component	Value	Description
<b>Possible Values</b>	0	<b>Undefined</b>
	<b>a</b>	<b>Official Close</b> – Explicit closing price value calculated and distributed by an exchange for the main trading session of a given trading day.
	b	<b>Official Indicative</b> – Exchange has provided an indicative price and marked it as indicative, however no trading activity is observed.
	c	<b>Official Carry Over</b> – Explicit Closing price value from a previous trading day carried forward by the exchange to the given trading day.
	d	<b>Last Price</b> – Final price disseminated by the exchange for the main trading session or dissemination period of a given trading day (for indices).
	<b>e</b>	<b>Last Eligible Price</b> – 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	<b>Manual</b> – Price disseminated manually (in case of production correction).

#### 2.3.2.4. PreviousInternalDailyClosingPriceType

The values of the quotation tag **PreviousInternalDailyClosingPriceType** conveyed on the NASDAQ BASIC 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 **PreviousInternalDailyClosingPriceType** is described in the table below (the values currently disseminated are highlighted in **green**):

**Table 11 InternalDailyClosingPriceType – technical implementation in QuantFEED®**

Component	Value	Description
<b>Tag Name</b>	PreviousInternalDailyClosingPriceType	FeedOS tag name.
<b>Numeric ID</b>	9156	FeedOS unique ID disseminated on S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
<b>Type</b>	Char	Char data type.
<b>Format</b>	<i>[Internal specific value]</i>	An <b>internal specific value</b> , detailing the type of daily closing price, as described below.



**Table 11 InternalDailyClosingPriceType – technical implementation in QuantFEED® (Continued)**

Component	Value	Description
<b>Possible Values</b>	0	<b>Undefined</b>
	a	<b>Official Close</b> – Explicit closing price value calculated and distributed by an exchange for the main trading session of a given trading day.
	b	<b>Official Indicative</b> – Exchange has provided an indicative price and marked it as indicative, however no trading activity is observed.
	c	<b>Official Carry Over</b> – Explicit Closing price value from a previous trading day carried forward by the exchange to the given trading day.
	d	<b>Last Price</b> – Final price disseminated by the exchange for the main trading session or dissemination period of a given trading day (for indices).
	e	<b>Last Eligible Price</b> – 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	<b>Manual</b> – Price disseminated manually (in case of production correction).

### 3. Official Closing Price

The closing price is the last trade price upon close, as provided by the exchange. If the instrument has an auction phase, the market sends the last auction price, which becomes the closing price.

### 4. Special Behavior – Extended Trading Hours

All instruments on the NASDAQ BASIC market data stream can be traded during extended trading hours. The behavior of these products during extended and regular hours represents a single trading session for a single business day (not a multisession behavior), as described in the table below:

**Table 12 Extended Trading Hours on NASDAQ BASIC market data stream (express in UTC)**

Business Day	Trading Hours	Time (UTC)	Signal	Trading Status Value
Current/Single	Early Trading Hours (TradingSessionId = -1)	04:00	open	17 – Ready to Trade
		09:30	close	
	Regular Trading Hours (TradingSessionId = 1)	09:30	OPEN	
		16:00	CLOSE	
	Late Trading Hours (TradingSessionId = -2)	16:00	open	
		20:00	close	

## 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](mailto:rts-support@spcapitaliq.com)
- Web: <http://support.quanthouse.com>.