S&P CAPITAL IQ
McGRAW HILL FINANCIAL

S&P Capital IQ Real-Time Solutions

FeedOS™ Feed Description

OMX NORDIC CASH

Reference n°: 20150805 – 26448 – 28048 – 28189

S&P Capital IQ Real-Time Solutions
FeedOS™ Feed Description: OMX NORDIC CASH
Reference 20150805 – 26448 – 28048 – 28189
August 05, 2015

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

United States
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

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

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

www.spcapitaliq.com

TABLE OF CONTENTS

FEEDOS™ OMX NORDIC CASH FEED DESCRIPTION

As part of the S&P Capital IQ Real-Time Solutions FeedOS™ documentation, this feed description provides you with details about the types of data disseminated on the OMX NORDIC CASH 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 OMX NORDIC CASH market data stream, in terms of:

1.1. Available Markets and Branches

This section details the list of Markets and Branches available on the OMX NORDIC CASH market data stream.

1.1.1. Markets

The OMX NORDIC CASH market data stream disseminates informations about the following markets:

Table 1 List of markets available on the OMX NORDIC CASH market data stream

FeedOS Market ID	Market
XCSE	Copenhagen Stock Exchange
XHEL	Helsinki Stock Exchange
XICE	Reykjavik Stock Exchange
XOSL	Oslo Bors
ISEC	Reykjavik Stock Exchange - ICEX Second Market
XSTO	Stockholm – OMX Nordic Exchange
XOPV	OMX OTC Publication Venue
XFND	Copenhagen – First North Denmark
XSAT	Stockholm – Aktietorget
FNSE	Stockholm – First North Stockholm
FNFI	Helsinki – First North Finland

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

```
MARKETS
market # 67     CC=DK/DENMARK/COPENHAGEN,DESCR=COPENHAGEN STOCK EXCHANGE,WEB=www.cse.dk
    MIC = XCSE
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 76     CC=FI/FINLAND/HELSINKI,DESCR=THE HELSINKI STOCK EXCHANGE,WEB=www.hex.com
    MIC = XHEL
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 113    CC=IS/ICELAND/REYKJAVIK,DESCR=ICELAND STOCK EXCHANGE,WEB=www.icex.is
    MIC = XICE
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 187    CC=NO/NORWAY/OLSO,DESCR=OSLO BORS,WEB=www.ose.no
    MIC = XOSL
    TimeZone = Europe/Oslo
    Country = NO
    NbMaxInstruments = 2000000
market # 379    CC=IS/ICELAND/REYKJAVIK,DESCR=ISEC (ICEX SECOND MARKET),WEB=www.isec.is
    MIC = ISEC
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 430    CC=SE/SWEDEN/STOCKHOLM,DESCR=OMX NORDIC EXCHANGE STOCKHOLM AB,WEB=
    MIC = XSTO
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
                                                                    (see next page)
```

```
MARKETS CONTINUED
market # 440    CC=SE/SWEDEN/STOCKHOLM,DESCR=OMX OTC PUBLICATION VENUE,WEB=www.omxgroup.com
    MIC = XOPV
    TimeZone = Europe/Stockholm
    Country = SE
    NbMaxInstruments = 2000000
market # 441    CC=DK/DENMARK/COPENHAGEN,DESCR=FIRST NORTH DENMARK,WEB=www.omxgroup.com/
firstnorth
    MIC = XFND
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 442    CC=SE/SWEDEN/STOCKHOLM,DESCR=AKTIETORGET,WEB=www.actietorget.se
    MIC = XSAT
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 443    CC=SE/SWEDEN/STOCKHOLM,DESCR=FIRST NORTH STOCKHOLM,WEB=www.omxgroup.com/
firstnorth
    MIC = FNSE
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
market # 492    CC=FI/FINLAND/HELSINKI,DESCR=FIRST NORTH
FINLAND,WEB=www.omxnordicexchange.com/firstnorth
    MIC = FNFI
    TimeZone =
    Country =
    NbMaxInstruments = 1000000
```

1.1.2. Branches

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

```
BRANCHES
    { XCSE CS   ESXXXX } qty: 171
    { XCSE NONE EXXXXX } qty: 2
    { XCSE NONE RMXXXX } qty: 586
    { XCSE WAR  RWXXXX } qty: 181
    { XHEL CS   ESXXXX } qty: 159
    { XHEL MF   EUXXSX } qty: 1
    { XHEL NONE RMXXXX } qty: 59
    { XHEL WAR  RWXXXX } qty: 3762
    { XHEL XCN  RWXXXX } qty: 2
    { XICE CORP DBXXXX } qty: 127
    { XICE CS   ESXXXX } qty: 18
    { XICE GO   DBXXXX } qty: 107
    { XICE MF   EUXXSX } qty: 2
    { XICE NONE RMXXXX } qty: 9
    { ISEC CS   ESXXXX } qty: 4
    { XSTO CB   DCXXXX } qty: 9
    { XSTO CS   ESXXXX } qty: 383
    { XSTO MF   EUXXSX } qty: 56
    { XSTO NONE RMXXXX } qty: 95
    { XSTO WAR  RWXXXX } qty: 13020
    { XSTO XCN  RWXXXX } qty: 1
    { XFND CS   ESXXXX } qty: 15
    { XSAT CB   DCXXXX } qty: 3
    { XSAT CS   ESXXXX } qty: 359
    { XSAT GO   DBXXXX } qty: 1
    { XSAT NONE RMXXXX } qty: 1
    { XSAT WAR  RWSXXX } qty: 1
    { FNSE CS   ESXXXX } qty: 552
    { FNSE NONE RMXXXX } qty: 1
    { FNFI CS   ESXXXX } qty: 17
```

1.2. Types of Instruments

The following sections describe the instruments available on the OMX NORDIC CASH 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 # 430/1010 = 901776370
    PriceCurrency               string{SEK}
    Symbol                      string{ATCO A}
    Description                 string{Atlas Copco AB ser. A}
    SecurityType                string{CS}
    FOSMarketId                 XSTO
    CFICode                     string{ESXXXX}
    RoundLot                    float64{1}
    MarketSegmentID             string{182}
    MarketSegmentDesc           string{STO Equities CCP}
    InternalCreationDate        Timestamp{2013-12-17 04:00:00:264}
    InternalModificationDate    Timestamp{2015-08-05 05:05:10:591}
    InternalSourceId            uint16{67}
    InternalEntitlementId       int32{1062}
    LocalCodeStr                string{45}
    ISIN                        string{SE0006886750}
    PriceIncrement_dynamic_TableId  uint32{4391026}
    OperatingMIC                string{XSTO}
    CCP_Eligible                bool{True}
```

1.2.2. Bonds

The sample below illustrates the details of a bond:

```
instr # 113/1084 = 236979260
    PriceCurrency               string{ISK}
    Symbol                      string{VOP 98 1}
    Description                 string{Vopnafjarðarhreppur 98 1}
    SecurityType                string{CORP}
    StdMaturity                 string{201902}
    FOSMarketId                 XICE
    CFICode                     string{DBXXXX}
    RoundLot                    float64{5000000}
    MarketSegmentID             string{24}
    MarketSegmentDesc           string{OMX ICE DP Fixed Income}
    InternalCreationDate        Timestamp{2013-12-17 04:00:00:331}
    InternalModificationDate    Timestamp{2015-08-05 05:05:10:605}
    InternalSourceId            uint16{67}
    InternalEntitlementId       int32{1062}
    LocalCodeStr                string{10669}
    ISIN                        string{IS0000004018}
    MaturityYear                uint16{2019}
    MaturityMonth               uint8{2}
    MaturityDay                 uint8{15}
    PriceIncrement_dynamic_TableId  uint32{4391020}
    OperatingMIC                string{XICE}
    CCP_Eligible                bool{False}
```

1.2.3. Rights

The sample below illustrates the details of a right:

```
instr # 430/1004 = 901776364
    PriceCurrency               string{SEK}
    Symbol                      string{BILI TO 1A}
    Description                 string{Bilia AB ser. A TO 1A}
    SecurityType                string{NONE}
    StdMaturity                 string{201512}
    StrikePrice                 float64{10}
    FOSMarketId                 XSTO
    ContractMultiplier          float64{1}
    CFICode                     string{RMXXXX}
    RoundLot                    float64{1}
    MarketSegmentID             string{1}
    MarketSegmentDesc           string{OMX STO Eq.rights\,Subs.Opt\, Int. shares}
    InternalCreationDate        Timestamp{2013-12-17 04:00:00:487}
    InternalModificationDate    Timestamp{2015-08-05 05:05:10:619}
    InternalSourceId            uint16{67}
    InternalEntitlementId       int32{1062}
    LocalCodeStr                string{62533}
    ISIN                        string{SE0002694497}
    MaturityYear                uint16{2015}
    MaturityMonth               uint8{12}
    MaturityDay                 uint8{28}
    PriceIncrement_dynamic_TableId  uint32{4391040}
    OperatingMIC                string{XSTO}
    CCP_Eligible                bool{False}
```

1.2.4. Warrants

The sample below illustrates the details of a warrant:

```
instr # 430/5472 = 901780832
    PriceCurrency               string{SEK}
    Symbol                      string{KB DJEU50UBS}
    Description                 string{Dow Jones Eurostoxx 50 KO (EUR 0\,01)}
    SecurityType                string{WAR}
    StrikePrice                 float64{0.01}
    FOSMarketId                 XSTO
    ContractMultiplier          float64{100}
    CFICode                     string{RWXXXX}
    RoundLot                    float64{1}
    MarketSegmentID             string{160}
    MarketSegmentDesc           string{OMX STO Certificates}
    InternalCreationDate        Timestamp{2013-12-17 04:00:00:460}
    InternalModificationDate    Timestamp{2015-08-05 05:05:10:615}
    InternalSourceId            uint16{67}
    InternalEntitlementId       int32{1062}
    LocalCodeStr                string{52903}
    ISIN                        string{CH0021115669}
    PriceIncrement_dynamic_TableId  uint32{4391012}
    OperatingMIC                string{XSTO}
    CCP_Eligible                bool{False}
```

1.3. Specific Referential Tags

The following sections describe specific referential tags available on the OMX NORDIC CASH market data stream:

- 1.3.1. MarketSegmentID
- 1.3.2. CCP_Eligible.

1.3.1. MarketSegmentID

The values of the referential tag **MarketSegmentID** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Referential* to detail the ID of the market segment.

FeedOS implementation of the tag `MarketSegmentID` is described 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.
Possible Values	`1`	OMX STO Eq. Rights, Subs. Opt, Int. Shares
	`2`	OMX STO Equities
	`3`	OMX STO Convertibles
	`4`	OMX STO Warrants
	`8`	First North STO
	`14`	OMX CPH Equities
	`16`	OMX CPH Investment Funds
	`17`	OMX STO Fund units
	`18`	MM Aktietorget-cotr
	`23`	OMX ICE Equities
	`24`	OMX ICE DP Fixed Income
	`26`	OMX ICE Collective Investm. Undertakings
	`52`	OMX ICE CP Fixed Income
	`55`	First North Convertibles STO
	`62`	OMX HEL Equities
	`63`	OMX HEL Equity rights
	`64`	OMX HEL Covered Warrants
	`69`	OMX HEL Fund units
	`85`	OMX ICE Fund units
	`100`	First North
	`101`	First North Iceland
	`104`	OMX CPH Other Collective Investments
	`110`	First North Finland

Table 2 MarketSegmentID – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	157	OMX HEL Exchange Traded Notes
	158	OMX STO Exchange Traded Notes
	159	OMX HEL Certificates
	160	OMX STO Certificates
	176	MM Aktietorget Conv
	180	OMX STO Equities NOK
	181	OMX HEL Equities CCP
	182	OMX STO Equities CCP
	183	OMX CPH Equities CCP
	185	OMX STO Fund units NOK
	195	First North NOK
	196	OMX CPH Exchange Traded Notes
	198	RänteTorget Bonds SEK
	199	OMX HEL Actively-Managed Funds
	200	OMX STO Actively-Managed Funds
	201	OMX HEL Equities intraday cross CCP
	202	OMX STO Equities intraday cross

1.3.2. CCP_Eligible

The values of the referential tag **CCP_Eligible** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Referential* to specify whether an instrument is cleared via the CCP or not.

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

Table 3 CCP_Eligible – technical implementation in FeedOS

Component	Value	Description
Tag Name	`CCP_Eligible`	FeedOS tag name.
Numeric ID	`9552`	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	`Bool`	Bool data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing whether an instrument is cleared via the CCP.
Possible Values	`True`	CCP eligibility and post trade anonymity.
	`False`	Default value, not sent.

2. Quotation Data

The following sections describe the characteristics of the quotation data on the OMX NORDIC CASH market data stream, in terms of:

- 2.1. Quotation Values

2.1. Quotation Values

The example below shows the possible values of an instrument on the OMX NORDIC CASH market data stream:

```
InstrumentStatusL1
-- 430/1278
        BID: 60 26776   @11
        ASK: 60.05      59149   @11
        LastPrice                       float64{60.05}
        LastTradeQty                    float64{2000}
        DailyHighPrice                  float64{60.55}
        DailyLowPrice                   float64{60}
        DailyTotalVolumeTraded          float64{329581}
        DailyTotalAssetTraded           float64{19855356.65}
        LastTradePrice                  float64{60.05}
        LastTradeTimestamp              Timestamp{2015-08-05 09:35:36:316}
        InternalDailyOpenTimestamp      Timestamp{2015-08-05 07:00:00:202}
        InternalDailyCloseTimestamp     Timestamp{2015-08-04 15:29:59:059}
        InternalDailyHighTimestamp      Timestamp{2015-08-05 07:14:38:174}
        InternalDailyLowTimestamp       Timestamp{2015-08-05 08:06:59:976}
        InternalPriceActivityTimestamp  Timestamp{2015-08-05 09:36:20:578}
        TradingStatus                   17=ReadyToTrade
        LastOffBookTradePrice           float64{60.2}
        LastOffBookTradeQty             float64{30}
        LastOffBookTradeTimestamp       Timestamp{2015-08-04 14:19:36}
        DailyOpeningPrice               float64{60.35}
        PreviousDailyTotalVolumeTraded  float64{1039033}
        PreviousDailyTotalAssetTraded   float64{62656272.1}
        PreviousDailyClosingPrice       float64{60.15}
        PreviousBusinessDay             Timestamp{2015-08-04}
        CurrentBusinessDay              Timestamp{2015-08-05}
        LastAuctionPrice                float64{60.4}
        LastAuctionVolume               float64{457}
        DailyTotalOffBookVolumeTraded   float64{0}
        DailyTotalOffBookAssetTraded    float64{0}
        LastAuctionImbalanceSide        char{S}
        LastAuctionImbalanceVolume      float64{4583}
        InternalDailyClosingPriceType   char{e}
        Footnotes                       string{0}
        InternalLastAuctionTimestamp    Timestamp{2015-08-05 06:59:57:035}
        MARKET_OMX_NORDIC_MarketSegmentState    string{T}
        MARKET_OMX_NORDIC_OrderBookTradingState string{T}
        MARKET_OMX_NORDIC_OrderBookHaltReason   string{}
```

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 OMX NORDIC CASH 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 4 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. For more details about Trading Status Kinematics, see also section 4.1. Management of the Trading Status Kinematics.
Possible Values	2	Trading Halt
	5	Price Indication
	15	New Price Indication
	17	Ready to Trade
	18	Not Available for Trading
	21	Pre-Open

2.3. Specific Quotation Tags

The following sections describe the specific quotation tags available on the OMX NORDIC CASH 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 available on the OMX NORDIC CASH market data stream:

- 2.3.1.1. TradeID
- 2.3.1.2. MARKET_OMX_NORDIC_OrderReferenceNumber
- 2.3.1.3. MARKET_OMX_NORDIC_TradeType
- 2.3.1.4. MARKET_OMX_NORDIC_NumberOfTrades
- 2.3.1.5. MARKET_OMX_NORDIC_CrossType
- 2.3.1.6. MARKET_OMX_NORDIC_ReportedTradeType.

2.3.1.1. TradeID

Each time a trade occurs, the values of the quotation tag **TradeID** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Context* to detail the trade identifier:

- 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	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the trade identifier.

2.3.1.2. MARKET_OMX_NORDIC_OrderReferenceNumber

Each time an order is given, the values of the quotation tag **MARKET_OMX_NORDIC_OrderReferenceNumber** conveyed on the OMX NORDIC CASH 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 `MARKET_OMX_NORDIC_OrderReferenceNumber` is described in the table below:

Table 6 MARKET_OMX_NORDIC_OrderReferenceNumber – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_OrderReferenceNumber`	FeedOS tag name.
Numeric ID	`16100`	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`	UInt32 data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the order reference number.

2.3.1.3. MARKET_OMX_NORDIC_TradeType

Each time a trade occurs, the values of the quotation tag **MARKET_OMX_NORDIC_TradeType** conveyed on the OMX NORDIC CASH 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_OMX_NORDIC_TradeType` is described in the table below:

Table 7 MARKET_OMX_NORDIC_TradeType – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_TradeType`	FeedOS tag name.
Numeric ID	`16101`	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	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the trade type.
Possible Values	`B`	Main Book
	`S`	Nordic@Mid

2.3.1.4. MARKET_OMX_NORDIC_NumberOfTrades

Each time a trade occurs, the values of the quotation tag **MARKET_OMX_NORDIC_NumberOfTrades** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Context* to detail the number of trades:

- 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_OMX_NORDIC_NumberOfTrades` is described in the table below:

Table 8 MARKET_OMX_NORDIC_NumberOfTrades – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_NumberOfTrades`	FeedOS tag name.
Numeric ID	`16102`	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`	UInt32 data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the number of trades.

2.3.1.5. MARKET_OMX_NORDIC_CrossType

Each time a cross trade occurs, the values of the quotation tag **MARKET_OMX_NORDIC_CrossType** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Context* to indicate the cross trade type:

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

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

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

Table 9 MARKET_OMX_NORDIC_CrossType – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_CrossType`	FeedOS tag name.
Numeric ID	16103	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	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the cross trade type.
Possible Values	`I`	Scheduled Intraday Auction
	`O`	Opening Cross
	`C`	Closing Cross
	`H`	Cross Halted for Securities

2.3.1.6. MARKET_OMX_NORDIC_ReportedTradeType

Each time a trade occurs, the values of the quotation tag **MARKET_OMX_NORDIC_ReportedTradeType** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Context* to indicate the reported trade type:

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

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

Table 10 MARKET_OMX_NORDIC_ReportedTradeType – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_ReportedTradeType`	FeedOS tag name.
Numeric ID	16104	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	`Char`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the reported trade type.
Possible Values	1	Regular
	2	Non-standard Settlement
	3	Exchange Granted Trade
	4	Weighted Average Price Trade
	5	Portfolio Trade
	6	Volume Weighted Average Trade
	7	Derivative Related Transaction
	8	Pre-opening Trade
	9	OTC

Table 10 MARKET_OMX_NORDIC_ReportedTradeType – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	A	OTC Non-standard
	B	SI Standard
	C	SI Non-standard
	D	Regular, OX Routed

2.3.2. Other Values

The following subsections describe other specific quotation tags on OMX NORDIC CASH market data stream:

- 2.3.2.1. InternalDailyClosingPriceType
- 2.3.2.2. Footnotes
- 2.3.2.3. MARKET_OMX_NORDIC_MarketSegmentState
- 2.3.2.4. MARKET_OMX_NORDIC_OrderBookTradingState
- 2.3.2.5. MARKET_OMX_NORDIC_OrderBookHaltReason.

2.3.2.1. InternalDailyClosingPriceType

The values of the quotation tag **InternalDailyClosingPriceType** conveyed on the OMX NORDIC CASH 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 11 InternalDailyClosingPriceType – technical implementation in FeedOS

Component	Value	Description
Tag Name	`InternalDailyClosingPriceType`	FeedOS tag name.
Numeric ID	`9155`	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	*`[Internal Specific Value]`*	An ***internal specific value***, detailing the type of daily closing price, as described below.

Table 11 InternalDailyClosingPriceType – technical implementation in FeedOS (Continued)

Component	Value	Description
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. Footnotes

The values of the quotation tag **Footnotes** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Other Values* to indicate Note Codes:

- 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 `Footnotes` is described in the table below:

Table 12 Footnotes – technical implementation in FeedOS

Component	Value	Description
Tag Name	`Footnotes`	FeedOS tag name.
Numeric ID	`9227`	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`	Char data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing Note Codes. **NOTE:** This field is a decimal encoded bit field where multiple Note Codes can be set concurrently by being binary OR:ed together. A value of zero means No Note Code Set. *Example 1:* `Footnotes=3` means that both NM and XR Note Codes are active. *Example 2:* `Footnotes=31` means that NM, XR, SP, PO and UD are set.

Table 12 Footnotes – technical implementation in FeedOS (Continued)

Component	Value	Description
Possible Values	0	No Note Codes available for the Orderbook.
	1	NM, New Market Company
	2	XR, Excluding participating in right/s
	4	SP, Excluding participating in split
	8	PO, Company subject to public offer
	16	UD, Under drawing
	32	SR, Excluding comb. Split and issue right/s
	64	UL, UnListed
	128	WI, When Issued
	256	BR, Company Bankruptcy
	512	SU, Suspension
	1024	RL, Removal from listing in process
	2048	SL, Other surveillance list reason
	4096	TO, A significant reverse takeover pending
	8192	CS, Cent shares
	16384	RS, Reversed Split
	32768	BS, Excluding comb. Bonus & Split
	65536	SS, Excluding comb. Split & Redemption share
	131072	FN, First North Company
	262144	OB, On the surveillance list
	524288	XD, Excluding dividend
	1048576	FE, Foreign non-EU/EEA Entity, excluding the Faroe Islands and Greenland
	2097152	SO, Sold-Out Buy-Back

2.3.2.3. MARKET_OMX_NORDIC_MarketSegmentState

Each time a change in the market segment occurs, the values of the quotation tag **MARKET_OMX_NORDIC_MarketSegmentState** conveyed on the OMX NORDIC CASH 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_OMX_NORDIC_MarketSegmentState` is described in the table below:

Table 13 MARKET_OMX_NORDIC_MarketSegmentState – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_MarketSegmentState`	FeedOS tag name.
Numeric ID	`14594`	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 market segment state.
Possible Values	`C`	Closed
	`I`	Scheduled Intraday Auction
	`L`	Closing Auction
	`O`	Opening Auction
	`P`	PreOpen
	`S`	PostTrade
	`T`	Continuous Trading

2.3.2.4. MARKET_OMX_NORDIC_OrderBookTradingState

The values of the quotation tag **MARKET_OMX_NORDIC_OrderBookTradingState** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the trading state of the order book:

- 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_OMX_NORDIC_OrderBookTradingState` is described in the table below:

Table 14 MARKET_OMX_NORDIC_OrderBookTradingState – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_OrderBookTradingState`	FeedOS tag name.
Numeric ID	`14595`	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 order book trading state.
Possible Values	`H`	Halted
	`T`	Trading on NASDAQ OMX Nordic
	`Q`	Auction Period

2.3.2.5. MARKET_OMX_NORDIC_OrderBookHaltReason

Each time an order book is no longer traded, the values of the quotation tag **MARKET_OMX_NORDIC_OrderBookHaltReason** conveyed on the OMX NORDIC CASH market data stream are disseminated via FeedOS data stream in *Other Values* to detail the halt reason:

- 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_OMX_NORDIC_OrderBookHaltReason` is described in the table below:

Table 15 MARKET_OMX_NORDIC_OrderBookHaltReason – technical implementation in FeedOS

Component	Value	Description
Tag Name	`MARKET_OMX_NORDIC_OrderBookHaltReason`	FeedOS tag name.
Numeric ID	`14596`	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 order book halt reason.
Possible Values	`TH`	Trading Halt
	`RH`	Regulatory Halt
	`MH`	Matching Halt
	`TS`	Technical Stop
	`KO`	Trading Halt - Knock-Out
	`VHD`	Volatility Halt – Dynamic
	`VHS`	Volatility Halt – Static
	`Space`	Reason Not Available

2.4. MBL and MBO Data*

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

3. Official Closing Price

For the instruments having an auction phase, the last auction price is used as the closing price. For all other instruments, the closing price is the last trade price upon close.

* The MBL and MBO 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. Special Behavior

The following sections detail the OMX NORDIC CASH market data stream special behavior in terms of:

- 4.1. Management of the Trading Status Kinematics
- 4.2. Management of the CLOSE Kinematics
- 4.3. Management of the Last Auction Tags.

4.1. Management of the Trading Status Kinematics

In the Level1 Market Data Kinematics **before 2014-12-01**, when the tag `MARKET_OMX_NORDIC_MarketSegmentState` disseminates the value `S=PostTrade`, the Trading Status is set to `18=Not Available for Trading`. Moreover, the exchange sends the CLOSE signal simultaneously with the value `C=Closed` of the tag `MARKET_OMX_NORDIC_MarketSegmentState`, while the Trading Status remains the same, as shown 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  15:24:14:122  159388634   *       *       9.45    49140@1 *       *
VU  15:25:00:000  159388634   MARKET_OMX_NORDIC_MarketSegmentState=S   TradingStatus=18
TE  15:25:02:673  159388634   *       *       *       *       !       0
TE  15:25:02:673  159388634   *       *       !       0       *       *
SI  16:00:00:000  159388634   CLOSE   9.5
TE  16:00:00:000  159388634   9.5     *       *       *       *       *       C
VU  16:00:00:000  159388634   MARKET_OMX_NORDIC_MarketSegmentState=C
```

In the Level1 Market Data Kinematics **after 2014-12-01**, when the tag `MARKET_OMX_NORDIC_MarketSegmentState` disseminates the value `S=PostTrade`, the Trading Status is set to `15=New Price Indication`. Moreover, the exchange sends the CLOSE signal before disseminating the value `C=Closed` on the tag `MARKET_OMX_NORDIC_MarketSegmentState`, while the Trading Status is set to `18=Not Available for Trading`, as shown 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  15:24:14:122  159388634   *       *       9.45    49140@1 *       *
VU  15:25:00:000  159388634   MARKET_OMX_NORDIC_MarketSegmentState=S   TradingStatus=15
SI  15:25:00:000  159388634   CLOSE   9.5
TE  15:25:00:000  159388634   9.5     *       *       *       *       *       C
TE  15:25:02:673  159388634   *       *       *       *       !       0
TE  15:25:02:673  159388634   *       *       !       0       *       *
VU  16:00:00:000  159388634   MARKET_OMX_NORDIC_MarketSegmentState=C   TradingStatus=18
```

4.2. Management of the CLOSE Kinematics

If trades occur during the trading day, the market sends the `CLOSE` signal and the closing price. However, if no eligible closing trade occurs during the trading day, the market sends only the `CLOSE` signal, without the closing price, as shown in the examples below:

Sample OMX CLOSE kinematics for a traded instrument

```
"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  15:34:11:046 140514971       *       *       *       *       !       0
TE  15:34:11:046 140514971       *       *       *       *       116.2   1000@1
TE  15:34:11:060 140514971       *       *       !       0       *       *
TE  15:34:11:060 140514971       *       *       115.7   1000@1  *       *
VU  15:55:00:002 140514971       MARKET_OMX_NORDIC_MarketSegmentState=L  TradingStatus=5
SI  15:59:37:144 140514971       CLOSE   113.3
TE  15:59:37:144 140514971       113.3   *       *       *       *       *       C
VU  15:59:37:144 140514971       TradingStatus=18
TE  15:59:37:144 140514971       *       *       !       0       *       *
TE  15:59:37:144 140514971       *       *       *       *       !       0
VU  15:59:59:957 140514971       MARKET_OMX_NORDIC_MarketSegmentState=S
VU  16:20:00:006 140514971       MARKET_OMX_NORDIC_MarketSegmentState=C
```

Sample OMX CLOSE kinematics for a not-traded instrument

```
"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  15:34:11:046 140514971       *       *       *       *       !       0
TE  15:34:11:046 140514971       *       *       *       *       116.2   1000@1
TE  15:34:11:060 140514971       *       *       !       0       *       *
TE  15:34:11:060 140514971       *       *       115.7   1000@1  *       *
VU  15:55:00:002 140514971       MARKET_OMX_NORDIC_MarketSegmentState=L  TradingStatus=5
VU  15:55:00:020 140514971       LastAuctionPrice=?      LastAuctionVolume=?
LastAuctionImbalanceSide=O      LastAuctionImbalanceVolume=?
SI  15:59:37:144 140514971       CLOSE   *
TE  15:59:37:144 140514971       *       *       *       *       *       *       C
VU  15:59:37:144 140514971       TradingStatus=18
TE  15:59:37:144 140514971       *       *       !       0       *       *
TE  15:59:37:144 140514971       *       *       *       *       !       0
VU  15:59:59:957 140514971       MARKET_OMX_NORDIC_MarketSegmentState=S
VU  16:20:00:006 140514971       MARKET_OMX_NORDIC_MarketSegmentState=C
```

4.3. Management of the Last Auction Tags

The Last Auction tags – `LastAuctionPrice` (**NumericID**: 9146, **Type**: `Float64`), `LastAuctionVolume` (**NumericID**: 9147, **Type**: `Float64`), `LastAuctionImbalanceSide` (**NumericID**: 9151, **Type**: `Char`) and `LastAuctionImbalanceVolume` (**NumericID**: 9152, **Type**: `Float64`) – are cleared at the beginning of each auction phase.

Moreover, each time the exchange sends a Net Order Imbalance Indicator (NOII) message with an auction reset, the Last Auction tags are reset, 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  02:04:16:691 140529444  MARKET_OMX_NORDIC_OrderBookTradingState=T
MARKET_OMX_NORDIC_OrderBookHaltReason=
TE  null    140529444       *       *       !       0       !       0
VU  07:00:00:001 140529444  MARKET_OMX_NORDIC_MarketSegmentState=P  TradingStatus=21
VU  08:30:00:001 140529444  MARKET_OMX_NORDIC_MarketSegmentState=O  TradingStatus=15
VU  08:30:00:013 140529444  LastAuctionPrice=?      LastAuctionVolume=?
LastAuctionImbalanceSide=O      LastAuctionImbalanceVolume=?
TE  08:38:05:590 140529444  *       *       100     1000@1  *       *
SI  08:45:00:023 140529444  OPEN    *
TE  08:45:00:023 140529444  *       *       *       *       *       *       O
VU  08:45:00:023 140529444  TradingStatus=17
VU  08:45:00:031 140529444  MARKET_OMX_NORDIC_MarketSegmentState=T
TE  09:17:21:598 140529444  *       *       101.6   1000@1  *       *
TE  09:17:37:760 140529444  *       *       *       *       102.5   1000@1
TE  09:53:52:929 140529444  *       *       101.7   874@1   *       *
TE  15:30:11:000 140529444  102.59  100000  *       *       *       *       f
Buyer=NDA,Seller=NDA,MARKET_OMX_NORDIC_ReportedTradeType=char{1}
TE  15:37:04:859 140529444  *       *       *       *       !       0
TE  15:37:19:757 140529444  *       *       *       *       103.5   1000@1
TE  15:37:25:149 140529444  *       *       102.6   1000@1  *       *
VU  15:55:00:002 140529444  MARKET_OMX_NORDIC_MarketSegmentState=L  TradingStatus=5
VU  15:55:00:021 140529444  LastAuctionPrice=?      LastAuctionVolume=?
LastAuctionImbalanceSide=O      LastAuctionImbalanceVolume=?
SI  15:59:58:098 140529444  CLOSE   *
TE  15:59:58:098 140529444  *       *       *       *       *       *       C
VU  15:59:58:098 140529444  TradingStatus=18
TE  15:59:58:098 140529444  *       *       !       0       *       *
TE  15:59:58:098 140529444  *       *       *       *       !       0
VU  15:59:59:957 140529444  MARKET_OMX_NORDIC_MarketSegmentState=S
VU  16:20:00:006 140529444  MARKET_OMX_NORDIC_MarketSegmentState=C
```

4.4. Microsecond Timestamp Precision on the Level1 Market Data

Effective **2015-03-23**, the server timestamps displays 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      11:00:22:091.520     901776370       *       *       *       *       12.42   1@1
TE      11:00:22:091.612     901776370       *       *       11.75   26@5    *       *
TE      11:00:22:091.612     901776370       *       *       *       *       6       942@39
TE      11:00:22:091.868     901776370       *       *       13.25   23@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.