S&P Capital IQ Real-Time Solutions

FeedOS™ Feed Description

ASX24 Feed

Reference n°: 20150213 – 21158 – 21953

S&P CAPITAL IQ
McGRAW HILL FINANCIAL

S&P Capital IQ Real-Time Solutions
FeedOS™ Feed Description: ASX24
Reference 20150213 – 21158 – 21953
February 24, 2015

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

US Offices
55 Water Street, 44th floor
New York, NY 10041
United States of America
Tel: +1-(212)-438-4346

130 East Randolph
One Prudential Plaza, Suite 2900
Chicago, IL 60601
United States of America
Tel: +1-(312)-233-7129

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

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

www.capitaliq.com

TABLE OF CONTENTS

FEEDOS™ ASX24 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 ASX24 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 ASX24 market data stream, in terms of:

1.1. Available Markets and Branches

This section details the list of Markets and Branches available on the ASX24 market data stream.

1.1.1. Markets

The ASX24 market data stream broadcasts informations about the following markets:

Table 1 List of markets available on the ASX24 market data stream

FeedOS Market ID	Market
XSFE	Sydney Futures Exchange Limited

The following example shows the complete list of markets available on the ASX24 market data stream and their IDs, returned by the `dumps` command:

```
MARKETS
market # 21     CC=AU/AUSTRALIA/SYDNEY,DESCR=SYDNEY FUTURES EXCHANGE
LIMITED,WEB=www.sfe.com.au
    MIC = XSFE
    TimeZone = Australia/Sydney
    Country = AU
    NbMaxInstruments = 2000000
```

1.1.2. Branches

The example below shows the complete list of branches available on the ASX24 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
    { XSFE FUT  FCXXXX } qty: 509
    { XSFE FUT  FFDXXX } qty: 18
    { XSFE FUT  FFFCXW } qty: 67
    { XSFE FUT  FFNXXX } qty: 34
    { XSFE MLEG FFXXXS } qty: 798
    { XSFE OPT  OCXDXX } qty: 1445
    { XSFE OPT  OCXNXX } qty: 540
    { XSFE OPT  OCXTXX } qty: 5223
    { XSFE OPT  OPXDXX } qty: 1445
    { XSFE OPT  OPXNXX } qty: 540
    { XSFE OPT  OPXTXX } qty: 5223
```

1.2. Types of Instruments

The following sections describe the instruments available on the ASX24 market data stream, according to their type:

- 1.2.1. Futures
- 1.2.2. Options
- 1.2.3. Multilegs.

1.2.1. Futures

The sample below illustrates the details of a future:

```
instr # 21/1003069 = 45043261
    PriceCurrency               string{AUD}
    Symbol                      string{IR}
    Description                 string{90 Day Bank Accepted Bill Futures}
    SecurityType                string{FUT}
    StdMaturity                 string{201409}
    FOSMarketId                 XSFE
    CFICode                     string{FFNXXX}
    MarketSegmentID             string{SFE}
    InternalCreationDate        Timestamp{2014-05-26 00:40:44:783}
    InternalModificationDate    Timestamp{2014-05-26 00:40:44:783}
    InternalSourceId            uint16{246}
    LocalCodeStr                string{237419}
    PriceIncrement_static       float64{0.01}
    MaturityYear                uint16{2014}
    MaturityMonth               uint8{9}
    MBLLayersDesc               string{0,1}
    OperatingMIC                string{XASX}
    SegmentMIC                  string{XSFE}
    FaceValue                   float64{1000000}
```

1.2.2. Options

The sample below illustrates the details of an option:

```
instr # 21/1003489 = 45043681
    PriceCurrency               string{AUD}
    Symbol                      string{IR}
    Description                 string{90 Day Bank Accepted Bill Options}
    SecurityType                string{OPT}
    StdMaturity                 string{201409}
    StrikePrice                 float64{98125}
    FOSMarketId                 XSFE
    CFICode                     string{OCXNXX}
    MarketSegmentID             string{SFE}
    InternalCreationDate        Timestamp{2014-05-26 00:40:52:614}
    InternalModificationDate    Timestamp{2014-05-26 00:40:52:614}
    InternalSourceId            uint16{246}
    LocalCodeStr                string{304028}
    PriceIncrement_static       float64{0.005}
    UnderlyingLocalCodeStr      string{237419}
    MaturityYear                uint16{2014}
    MaturityMonth               uint8{9}
    MBLLayersDesc               string{0,1}
    OperatingMIC                string{XASX}
    SegmentMIC                  string{XSFE}
    FaceValue                   float64{1000000}
```

1.2.3. Multilegs

The sample below illustrates the details of a multileg:

```
instr # 21/1021784 = 45061976
    SecurityType                string{MLEG}
    StdMaturity                 string{201409}
    FOSMarketId                 XSFE
    CFICode                     string{FFXXXS}
    NbLegs                      uint8{2}
    MarketSegmentID             string{SFE}
    InternalCreationDate        Timestamp{2014-06-19 07:45:00:095}
    InternalModificationDate    Timestamp{2014-06-19 07:45:00:095}
    InternalSourceId            uint16{246}
    LocalCodeStr                string{330637}
    PriceIncrement_static       float64{1}
    MaturityYear                uint16{2014}
    MaturityMonth               uint8{9}
    MBLLayersDesc               string{0,1}
    OperatingMIC                string{XASX}
    SegmentMIC                  string{XSFE}
    LegFOSInstrumentCode        uint32{45049057}
    LegFOSInstrumentCode_1      uint32{45061975}
    LegRatioQty                 float64{1}
    LegRatioQty_1               float64{1}
```

1.3. Specific Referential Tags

The following sections detail the specific referential tags available on the ASX24 market data stream:

1.3.1. Description

The values of the referential tag **Description** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Referential* to characterize an instrument.

FeedOS implementation of the values currently available for the tag `Description` is detailed in the table below:

Table 2 Description – technical implementation in FeedOS

Component	Value	Description
Tag Name	`Description`	FeedOS tag name.
Numeric ID	107	FeedOS unique ID disseminated on 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***, characterizing the instrument.

1.3.2. Factor

The values of the referential tag **Factor** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Referential* to specify the Contract Value Factor by which a price must be adjusted to determine the true nominal value of one futures/options contract.

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

Table 3 Factor – technical implementation in FeedOS

Component	Value	Description
Tag Name	`Factor`	FeedOS tag name.
Numeric ID	`228`	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***, specifying the Contract Value Factor by which a price must be adjusted to determine the true nominal value of one futures/options contract. **NOTE:** The FeedOS tag `Factor` replaces the tag `RoundLot`, when the Financial Type is a Commodity or an Equity.

1.3.3. FaceValue

The values of the referential tag **FaceValue** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Referential* to specify the amount of money stated on the face of a note, bond or stock.

FeedOS implementation of the tag `FaceValue` is detailed in the table below:

Table 4 Description – technical implementation in FeedOS

Component	Value	Description
Tag Name	`FaceValue`	FeedOS tag name.
Numeric ID	`9565`	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*** specifying the amount of money stated on the face of a note, bond or stock. **NOTE:** The FeedOS tag `FaceValue` replaces the tag `RoundLot`, when the Financial Type is a Government Bond or a Bank Bill.

1.3.4. SecurityTradingID

The values of the referential tag **SecurityTradingID** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Referential* to specify the instrument Security Code.

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

Table 5 SecurityTradingId – technical implementation in FeedOS

Component	Value	Description
Tag Name	`SecurityTradingId`	FeedOS tag name.
Numeric ID	`9525`	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	*`[CCMYY[C999999[o]]]`*	CC – Commodity Code (Future OR Option OR Spread) M – Month YY – Year C – Call/Put Indicator 999999 – Option Strike and o – Overnight Option Indicator where the Month codes are: F – January G – February H – March J – April K – May M – June N – July Q – August U – September V – October X – November Z – December
Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, detailing the instrument Security Code.

2. Quotation Data

The sections below describe the characteristics of the quotation data on the ASX24 market data stream, in terms of:

2.1. Quotation Values

The example below shows the possible values of an instrument on the ASX24 market data stream:

```
InstrumentStatusL1
-- 21/1003069
        BID: 97.34      11035   @57
        ASK: 97.35      6235    @50
        LastPrice                       float64{97.35}
        LastTradeQty                    float64{1}
        DailyHighPrice                  float64{97.35}
        DailyLowPrice                   float64{97.35}
        DailyTotalVolumeTraded          float64{181}
        DailyTotalAssetTraded           float64{17620.35}
        LastTradePrice                  float64{97.35}
        LastTradeTimestamp              Timestamp{2014-06-26 09:57:30:809}
        InternalDailyOpenTimestamp      Timestamp{2014-06-26 07:07:30:052}
        InternalDailyCloseTimestamp     Timestamp{2014-06-26 06:30:00:076}
        InternalDailyHighTimestamp      Timestamp{2014-06-26 07:13:29:217}
        InternalDailyLowTimestamp       Timestamp{2014-06-26 07:13:29:217}
        InternalPriceActivityTimestamp  Timestamp{2014-06-26 09:58:59:078}
        TradingStatus                   17=ReadyToTrade
        TradingSessionId                int8{2}
        SessionTotalOffBookAssetTraded  float64{0}
        SessionTotalOffBookVolumeTraded float64{0}
        SessionTotalVolumeTraded        float64{181}
        SessionOpeningPrice             float64{97.35}
        PreviousSessionClosingPrice     float64{97.34}
        SessionHighPrice                float64{97.35}
        SessionLowPrice                 float64{97.35}
        SessionTotalAssetTraded         float64{17620.35}
        DailyOpeningPrice               float64{97.35}
        PreviousDailyTotalVolumeTraded  float64{9654}
        PreviousDailyTotalAssetTraded   float64{939855.33}
        PreviousDailyClosingPrice       float64{97.34}
        PreviousBusinessDay             Timestamp{2014-06-26}
        CurrentBusinessDay              Timestamp{2014-06-27}
        PreviousDailySettlementPrice    float64{97.35}
        InternalLastAuctionTimestamp    Timestamp{2014-06-26 06:58:00:749}
        PriceActivityMarketTimestamp    Timestamp{2014-06-26 09:58:59:036}
        SettlementPriceDate             Timestamp{2014-06-26}
```

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 ASX24 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 6 Trading Status of the ASX24 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	*`[Exchange Specific Value]`*	An ***exchange specific value***, as described below, concerning the characteristics of the trading status.
Possible Values	`5`	Price Indication
	`16`	Trade Dissemination Time
	`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 ASX24 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 ASX24 market data stream:

- 2.3.1.1. TradeCondition.

2.3.1.1. TradeCondition

The values of the quotation tag **TradeCondition** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Context* to identify the a particular condition applicable to the 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.

QuantFEED® implementation of the tag `TradeCondition` is described in the table below:

Table 7 TradeCondition – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`TradeCondition`	FeedOS tag name.
Numeric ID	`277`	FeedOS unique ID disseminated on 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 particular condition applicable to the trade.
Possible Values	`<Empty>`	Normal Trade, trade price matches both order prices. Default value, not sent.
	`X`	Crossed
	`Z`	Sweeping Trade, trade price matches resting order
	`Z-X`	Sweeping Trade, trade price matches resting order – Crossed
	`R`	Levelling Trade, trade price may be different than price of resting order(s)
	`R-X`	Levelling Trade, trade price may be different than price of resting order(s) – Crossed
	`1`	Spread-to-Underlying trade (price is based on the order of the underlying future relating to intra, inter or custom matching with an outright order)
	`1-X`	Spread-to-Underlying trade (price is based on the order of the underlying future relating to intra, inter or custom matching with an outright order) – Crossed
	`AA-4`	Intra-Spread-to-Intra-Spread trade (price is based on the near contract's prior day settlement)
	`AA-4-X`	Intra-Spread-to-Intra-Spread trade (price is based on the near contract's prior day settlement) – Crossed
	`3`	Inter-Spread-to-Inter-Spread trade (price is based on the secondary's contract's prior day settlement)
	`3-X`	Inter-Spread-to-Inter-Spread trade (price is based on the secondary's contract's prior day settlement) – Crossed
	`AH`	Custom-to-Custom trade
	`AH-X`	Custom-to-Custom trade – Crossed

2.3.2. Other Values

The following subsections describe the other values available on the ASX24 market data stream:

2.3.2.1. LowLimitPrice

The values of the quotation tag **LowLimitPrice** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the low limit of a 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 `LowLimitPrice` is described in the following table:

Table 8 LowLimitPrice – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`LowLimitPrice`	FeedOS tag name.
Numeric ID	`1148`	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***, indicating the low limit of a price.

2.3.2.2. HighLimitPrice

The values of the quotation tag **HighLimitPrice** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the high limit of a 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 `HighLimitPrice` is described in the following table:

Table 9 HighLimitPrice – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`HighLimitPrice`	FeedOS tag name.
Numeric ID	`1149`	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***, indicating the high limit of a price.

2.3.2.3. TradingReferencePrice

The values of the quotation tag **TradingReferencePrice** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the reference 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 `TradingReferencePrice` is described in the following table:

Table 10 TradingReferencePrice – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`TradingReferencePrice`	FeedOS tag name.
Numeric ID	`9370`	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***, indicating the reference price.

2.3.2.4. SettlementPriceDate

The values of the quotation tag **SettlementPriceDate** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the date of the settlement 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 `SettlementPriceDate` is described in the following table:

Table 11 SettlementPriceDate – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`SettlementPriceDate`	FeedOS tag name.
Numeric ID	`9380`	FeedOS unique ID disseminated on S&P Capital IQ Real-Time Solutions data stream. This is the numeric equivalent of the tag name.
Type	`Timestamp`	Timestamp data type.
Format / Possible Values	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating the date of the settlement price.

2.3.2.5. SettlementPriceType

The values of the quotation tag **SettlementPriceDate** conveyed on the ASX24 market data stream are disseminated via FeedOS data stream in *Other Values* to indicate the type of settlement 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 `SettlementPriceType` is described in the following table:

Table 12 SettlementPriceType – technical implementation in QuantFEED®

Component	Value	Description
Tag Name	`SettlementPriceType`	FeedOS tag name.
Numeric ID	`9383`	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`	Timestamp data type.
Format	*`[Exchange Specific Value]`*	An ***exchange specific value***, indicating the type of settlement price.
Possible Values	a	Official Daily Settlement Price
	b	Official Indicative Settlement Price

2.4. MBL, MBO and BBO Data*

The MBL and MBO books are full depth.

3. Official Closing Price

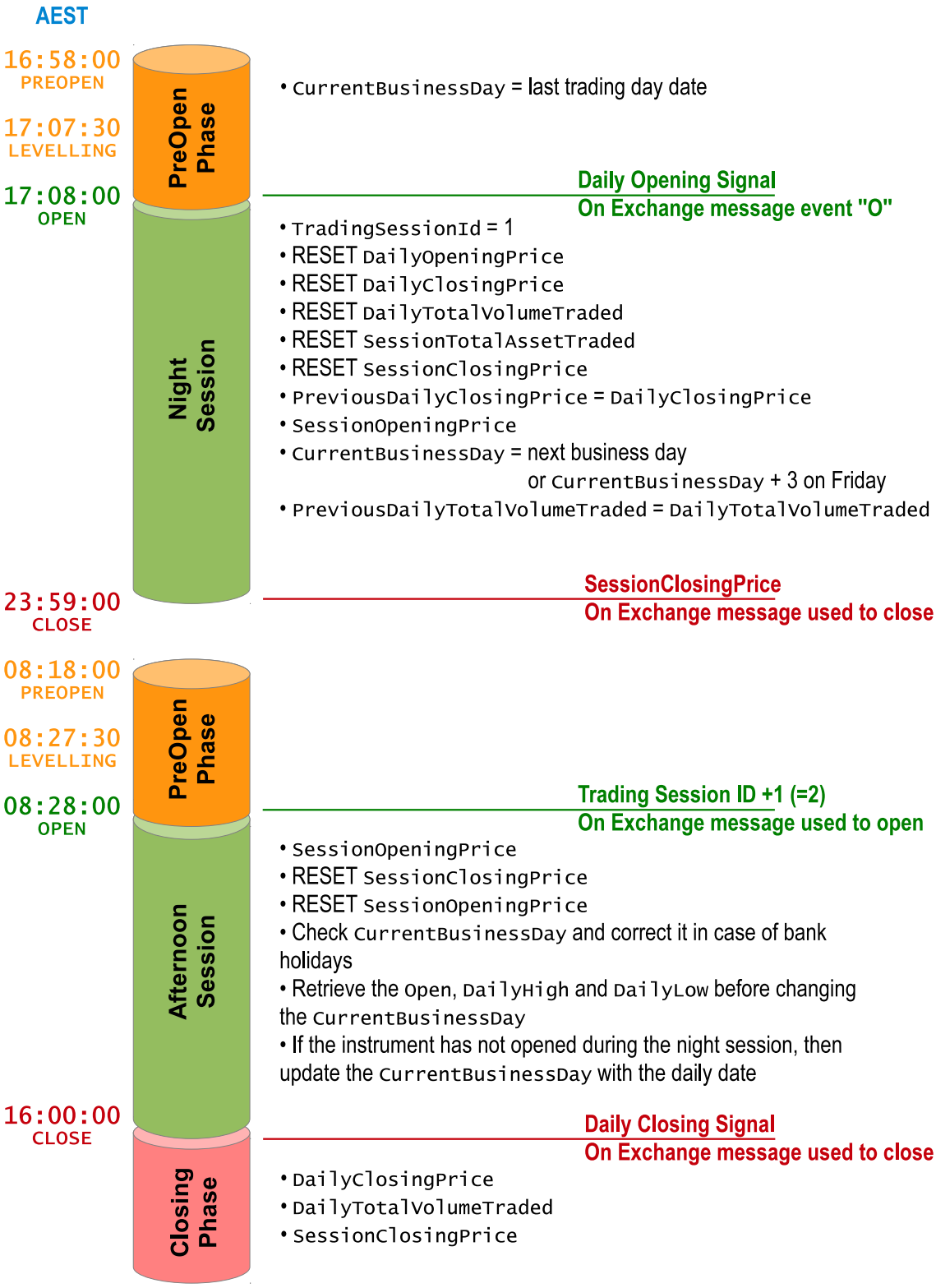
The closing price is the last trade price upon close. The settlement price is handled when provided by the market.

4. Multi-Session Kinematics

The following diagram describes the main trading phases and the update mechanism of the tags on the ASX24 market data stream:

* 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.

Figure 1 Example of tags update mechanism for a 90 Day Bank Bill (IR) on the ASX24 market data stream in Australian Eastern Standard Time



5. Special Behavior

The following section detail the ASX24 market data stream special behavior in terms of:

- 5.1. Management of the Best Bid/Best Ask in Level1 Market Data
- 5.2. Management of the Volatility Controls
- 5.3. Market News.

5.1. Management of the Best Bid/Best Ask in Level1 Market Data

The format of the Level1 Market Data merges the outright and implied orders in the Best Bid and Best Ask, as shown below:

```
     Outright             Implied                 L1 After
   Bid  |  Ask          Bid  |  Ask              Bid  |  Ask
-----------------    -----------------   ==>   ------------------
100 x 3 | 101 x 2    100 x 1 |   --            100 x 4 | 101 x 2
```

5.2. Management of the Volatility Controls

ASX DERIVATIVES Exchange uses several Volatility Controls – such as the *Anomalous Order Threshold (AOT)* and the Session State '*Regulatory Halt*' – on the following Equity Index Futures contracts:

- ASX SPI 200™ index futures (AP)
- S&P/ASX 200 Resources index futures (AR)
- S&P/ASX 200 Financials-x-A-REIT index futures (AF).

The **Anomalous Order Threshold** (AOT) prevents aggressive orders from entering the market outside an allowed range, based on a Dynamic Reference Price (as described in section 2.3.2.3. TradingReferencePrice). The AOT range will initially be set at +/- 0.5% from the AOT reference price (as described in sections 2.3.2.1. LowLimitPrice and 2.3.2.2. HighLimitPrice). The AOT reference price is a moving average and is recalculated at regular intervals.

The **Regulatory Halt Session State** re-sets the AOT reference price in the event of erroneous trading. If a Regulatory Halt occurs, the Trading Status is set to `5=Price Indication` (as described in section 2.2. Trading Status) and any spread orders and custom orders related to the halted product are purged.

5.3. Market News

Each time the exchange sends a text message to inform about ad hoc events, this is disseminated in the ASX24 market data stream in *Market News*:

- in the callback carrying the Level1 event `notif_MarketNews()`, for C++
- in the event handler `MarketNewsEventHandler`, for C#
- in the callback carrying the Level1 event `quotNotifMarketNewsEvent`, for Java.

The Market News messages on ASX24 market data stream notify about:

- General announcements
- EFP trade information
- Block trade information
- Central strike option volatility
- Single Session Option series underlying settlement price
- Serial Option underlying settlement price
- Custom RFQs (from participants)
- Settlement information that is normally distributed on the last trading date to the market as a text message, for Index and Non-Index product types.

Below is an example showing the message dissemination in the Market News:

```
“MN : MarketNews”

MN      null    2015-06-17 10:44:01:974 XSFE    Normal  Message from EFP       EFP Regd @
16:28 339 YTH5 @ 98.030      related_instruments: 21/1035797 21/1035796
MN      null    2015-06-18 00:58:35:971 XSFE    Normal  Message from EFP       Block Trade
Reg'd: YTH5 25 Lots @ 41.65 related_instruments: 21/1035797 21/1035796
```

6. 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: http://support.quanthouse.com.