S&P Capital IQ's Real-Time Solutions

QuantFEED Extended MBL Guide

Document Version: 1.3

Released: December 09, 2013

S&P CAPITAL IQ
McGRAW HILL FINANCIAL

S&P Capital IQ's Real-Time Solutions (QuantHouse®) – QuantFEED®
QuantFEED Extended MBL Guide
Document Version: 1.3
Released December 09, 2013

Corporate Headquarters
S&P Capital IQ's Real-Time Solutions (QuantHouse®)
52 Rue de la Victoire
75009 Paris
France
Tel: +33 (0) 1 73 02 32 11
Fax: +33 (0) 1 73 02 32 12

UK Office
10 Foster Lane
London EC2V 6HR
United Kingdom
Tel: +44 (0) 203 107 1676

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

www.quanthouse.com

Disclaimer for Technical Documents

QuantHouse® S.A.S. endeavors to include accurate and current information in its materials. However, QuantHouse® does not warrant the accuracy or completeness of the information contained herein. QuantHouse® may change such information at any time, but makes no commitment to update it.

References by QuantHouse® to products offered by third-parties do not constitute an endorsement by QuantHouse® of such products and should not be construed as an association with their owners.

YOUR USE OF THE INFORMATION HEREIN IS AT YOUR OWN RISK. SUCH INFORMATION IS PROVIDED ON AN "AS IS" BASIS. QUANTHOUSE® S.A.S. MAKES NO REPRESENTATION, UNDERTAKES NO OBLIGATION, AND PROVIDES NO WARRANTY OF ANY KIND WITH RESPECT TO THE INFORMATION CONTAINED HEREIN, WHETHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. IF YOU CHOOSE TO USE SUCH INFORMATION, YOU ARE ACKNOWLEDGING THAT YOU HAVE READ THIS DISCLAIMER, UNDERSTAND IT, AGREE TO ABIDE BY, AND BE BOUND BY, ITS PROVISIONS.

Use of the Information

The information constitutes proprietary material and is either owned by or licensed to QuantHouse®. Further, it is protected by intellectual property rights. No information may be used, reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise) or for any purpose, except as licensed expressly by QuantHouse® S.A.S.

Trademarks

QUANTHOUSE®, the QuantHouse® logo and product names are trademarks of QuantHouse® S.A.S. and QuantHouse® S.A.S. reserves all intellectual property rights with respect to the trademarks. All other trademarks are the trademarks of their respective owners.

Copyright

Contents

1 About This Guide

As part of S&P Capital IQ's Real-Time Solutions QuantFEED documentation set, this guide provides you with an outline of the underlying structure of an MBL subscription request. It also details the associated messages, event definition and examples.

1.1 Who This Guide Is For

This document is primarily intended for the use of QuantFEED software engineers, developers and other team members using QuantFEED. Furthermore, this guide also addresses issues and topics concerning any person who plans to develop software that interacts with QuantFEED middleware technology in particular.

1.2 What Do You Need to Use This Guide

To integrate the market data stream into your applications, general knowledge of market data acquisition and deployment is mandatory. Moreover, a good understanding of financial markets and instruments, including standards and protocols, is recommended as well.

General knowledge of **Application Programming Interface** (API) functioning and specifications for routines, data structures, object classes and variables is also required. Depending on the platform you use—C++, C# or Java—, good coding skills are also necessary.

1.3 Conventions Used in This Guide

This guide uses several common conventions to help you locate and interpret information easily. Following is a summary of the typographical conventions:

- Concepts and new terms are in **bold**.
- Sample source code—including keywords and variables—, and text that you should type appear in `monospaced font`.
- Folder, file, field names and placeholders are *italicized*.
- Menu names and commands appear in **bold**, and menu commands are separated by ">".
- In numbered steps, commands or options that you need to click or select are shown in **boldface**.

In addition to typographical conventions, the following special elements are included to set off different types of information to make them easily recognizable:

Note

Notes provide some additional details or information related to the topic that might help you expand your knowledge or understanding.

Tip

Tips offer helpful shortcuts or easier ways to do something.

Caution!

Cautions provide recommended details or other important information you need to know about consequences of using a feature or executing a task.

Warning

Warnings give you essential details or data about indispensable conditions or settings of your system, third-party applications and mandatory steps you should observe and apply on your platform, procedures, practices and tasks.

1.4 Document History

The table below summarizes the changes in the previous versions of this guide:

Release Date	Version	Change log
2013-12-09	1.3	Update of the diagrams. Additional details about `Continuation Flag`.
2011-11-18	1.2	New examples for orderbook update.
2010-12-10	1.1	New diagram of client/server messages.
2010-11-08	1.0	Creation of the document.

1.5 We Would Like to Hear from You

As the reader of this document and user of our products, you are our most important critic and commentator. We appreciate your opinion and want to know what you like about our work, what you dislike, what we could do better, what topics you would like to see us cover, but also any other comments and suggestions you wish to share with S&P Capital IQ’s Real-Time Solutions.

You can e-mail or write us to let us know what you did or did not find useful about this guide, as well as other topics and details you would like us to cover in the subsequent releases. When you write, please be sure to include this document’s title and version, as they appear on the copyright page. We will carefully review your comments and share them with the authors and contributors who worked on this guide.

E-mail:
support@quanthouse.com

Mail:
S&P Capital IQ's Real-Time Solutions (QuantHouse ®)
52 Rue de la Victoire
75009 Paris
France

2 QuantFEED Extended MBL Guide

The **Market-By-Level** (MBL) refers to the top best prices in the market with all the orders at the same price being aggregated. The request `SubscribeInstrumentsMBL` can carry several order books (or “order book layers”) for a given instrument. Currently, most feed handlers disseminate data only on layer # 0.

The following sections detail the MBL mechanism and its current implementation:

- Understanding Extended MBL
- MBL Related Messages
- Examples of Order Book Updates
- Implementation Recommendations.

2.1 Understanding Extended MBL

When implied quantities are available from the exchange, S&P Capital IQ’s Real-Time Solutions’s feed handlers merge them with the outright quantities. The result is a single order book, aggregated by price, where it is not possible to distinguish between implied and outright orders. In this case, if you use `SubscribeInstrumentsMBL`, you receive a single order book as layer # 0.

In an upcoming implementation of the multilayer support, S&P Capital IQ’s Real-Time Solutions’s feed handlers will be able to generate more layers, for instance:

- layer # 0: outright orders
- layer # 1: implied orders
- layer # 2: retail orders
- layer # 3: odd lot orders
- layer # 4: term orders.

Subsequently, the semantics of layer # 0 will change, from “default order book” to “outright orders only”. If you didn’t implement the multilayer support, the impact will be minimal—the order book quantities will appear smaller.

For consolidated instruments, such as equity feeds, each layer will carry the order book of the input feeds. For instance, when consolidating US equity feeds, you could have five layers corresponding to NYSE, NASDAQ, ARCA, BATS-US and DirectEdge. The number of layers is open and depends in this case on the server-side configuration.

When compared to Level 2 subscriptions, `SubscribeInstrumentsMBL` also provides you with:

- **Two Timestamps**—the official Market Timestamp (which can be null) and the Server Timestamp.
- **Number of Orders** (when provided by the exchange)—the new MBL is designed to host order books that detail the price, the cumulated quantity and the number of orders.
- **Other Values**—the extended MBL is able to carry “other values” (currently used only on consolidated feeds).

The extended `MBLDeltaRefresh` event contains a new kind of Action `BidRemoveLevelAndAppend` (respectively `AskRemoveLevelAndAppend`) that deletes a level and valuates the last visible limit of the book, in a single event.

Moreover, the `MBLDeltaRefresh` event contains a boolean flag named `ContinuationFlag`. This flag signals that the resulting cache book is not consistent (for example, the book may be crossed) until the first delta refresh is received with a false `ContinuationFlag`.

2.2 MBL Related Messages

The MBL-related messages are:

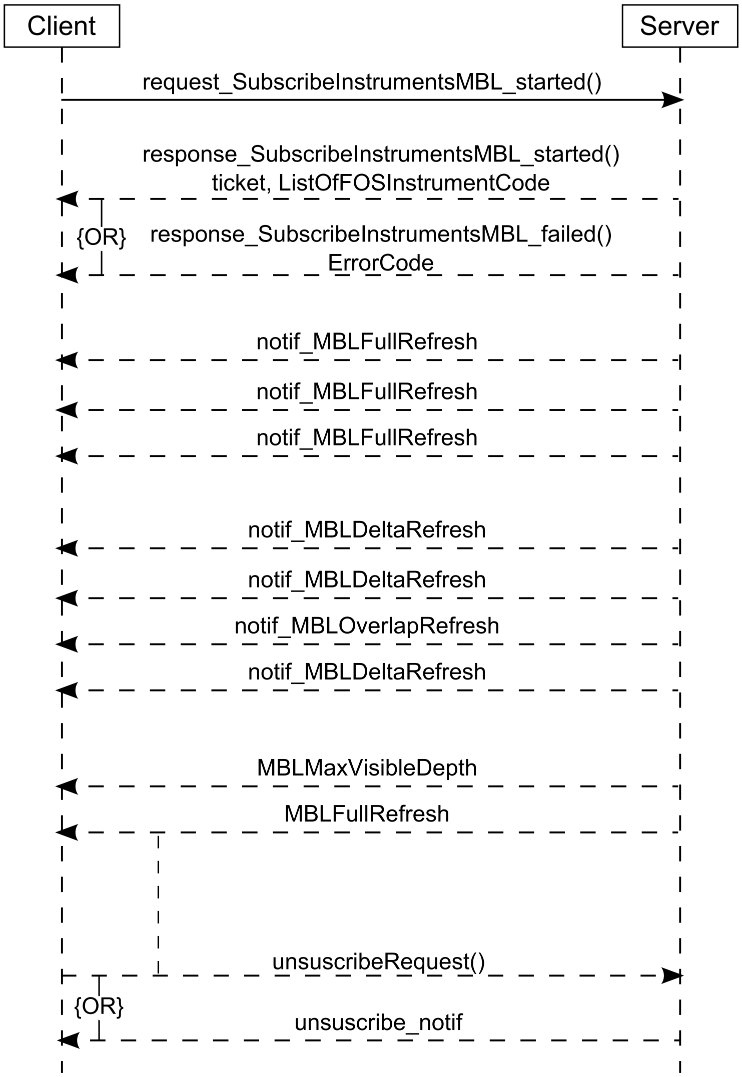
- SubRequest, unSubRequest
- SubResponse
- Notifications.

The following sections describe:

- MBL Sequence Diagram
- MBL Event Definition

2.2.1 MBL Sequence Diagram

When you subscribe to the extended MBL, the initial snapshot is asynchronous (not part of the response message), as shown in the diagram below:



Extended MBL sequence diagram

2.2.2 MBL Event Definition

`MBLDeltaRefresh` performs a small update on a given layer. The type of update is indicated by the field named `Action`. The table below details the supported actions:

Table 01—List of supported actions

Action	Description
BidChangeQtyAtLevel	Updates the bid quantity at the specified level.
AskChangeQtyAtLevel	Updates the ask quantity at the specified level.
ALLClearFromLevel	Deletes ALL entries, starting from Level (included).
BidClearFromLevel	Deletes bid entries, starting from Level (included).
AskClearFromLevel	Deletes ask entries, starting from Level (included).
BidInsertAtLevel	Inserts bid entry (Price, Quantity) at Level and shifts subsequent lines down.
AskInsertAtLevel	Inserts ask entry (Price, Quantity) at Level and shifts subsequent lines down.
BidRemoveLevel	Removes bid entry at Level and shifts subsequent lines up.
AskRemoveLevel	Removes ask entry at Level and shifts subsequent lines up.
BidRemoveLevel AndAppend	Removes the bid entry at the specified level and appends a new bid entry at the end of the order book.
AskRemoveLevel AndAppend	Removes the ask entry at the specified level and appends a new ask entry at the end of the order book.

`MBLOverlapRefresh` updates a part or the entire Bid and/or Ask sides on a given layer.

`MBLFullRefresh` massively updates order books (all layers of several instruments). When you start an MBL subscription, this notification conveys the "initial snapshots". You can also use it when you want to fully refresh one or more instruments.

`MBLMaxVisibleDepth` updates the "visible" depth (how many top prices are visible to subscribers). `-1` indicates that the depth is not limited.

> **Note**
>
> In `MBLFullRefresh`, the `MaxVisibleDepth` is part of the initial snapshot.

2.3 Examples of Order Book Updates

The following sections provide you with several examples concerning the order book updates. The actions `OrderBookDeltaAction AskXXX` have the same behavior on the Ask side as the `OrderBookDeltaAction BidXXX` on the Bid side.

The following QuantFEED MBL convention applies:

- The `()` denote a valuable list of the same type
- The `{}` denote a structure made by several fields.

The list of examples include:

- MBLDeltaRefresh, action=BidChangeQtyAtLevel
- MBLDeltaRefresh, action=BidRemoveLevel

- MBLDeltaRefresh, action=BidInsertAtLevel
- MBLDeltaRefresh, action=BidRemoveLevelAndAppend
- MBLDeltaRefresh, action=BidClearFromLevel
- MBLDeltaRefresh, action=ALLClearFromLevel
- Continuation Flag
- MBLOverlapRefresh
- MBLMaxVisibleDepth.

2.3.1 MBLDeltaRefresh, action=BidChangeQtyAtLevel

2.3.1.1 Before

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3100 x 2331 @ 10 ASK 53.3500 x 1936 @ 5
2 BID 53.3000 x 8109 @ 8  ASK 53.3600 x 2978 @ 7
3 BID 53.2900 x 2482 @ 12 ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.1.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:40:519,
              Market_utc_time=2010-11-08 16:22:40:519}
- Action=BidChangeQtyAtLevel
- Level=0
- Price=<not relevant>
- Qty={cumulated=958, nb_orders=3}
- ContinuationFlag=false
- OtherValues=()
```

2.3.1.3 After

```
0 BID 53.3200 x 958  @ 3  ASK 53.3400 x 501  @ 3
1 BID 53.3100 x 2331 @ 10 ASK 53.3500 x 1936 @ 5
2 BID 53.3000 x 8109 @ 8  ASK 53.3600 x 2978 @ 7
3 BID 53.2900 x 2482 @ 12 ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.2 MBLDeltaRefresh, action=BidRemoveLevel

2.3.2.1 Before

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3100 x 2331 @ 10 ASK 53.3500 x 1936 @ 6
2 BID 53.3000 x 8109 @ 8  ASK 53.3600 x 2978 @ 7
3 BID 53.2900 x 2482 @ 12 ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.2.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:40:887,
              Market_utc_time=2010-11-08 16:22:40:887}
- Action=BidRemoveLevel
- Level=1
- Price=<not relevant>
- Qty={cumulated=<not relevant>, nb_orders=<not relevant>}
- ContinuationFlag=false
- OtherValues=()
```

2.3.2.3 After

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3000 x 8109 @ 8  ASK 53.3500 x 1936 @ 6
2 BID 53.2900 x 2482 @ 12 ASK 53.3600 x 2978 @ 7
3 BID ******************* ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.3 MBLDeltaRefresh, action=BidInsertAtLevel

2.3.3.1 Before

```
0 BID 53.3000 x 8109 @ 8  ASK 53.3500 x 1936 @ 6
1 BID 53.2900 x 2482 @ 12 ASK 53.3600 x 2978 @ 7
2 BID 53.2800 x 3829 @ 8  ASK 53.3700 x 1988 @ 7
3 BID ******************* ASK 53.3800 x 814  @ 6
4 BID ******************* ASK *******************
```

2.3.3.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:41:807,
              Market_utc_time=2010-11-08 16:22:41:807}
- Action=BidInsertAtLevel
- Level=2
- Price=53.285
- Qty={cumulated=1914, nb_orders=4}
- ContinuationFlag=false
- OtherValues=()
```

2.3.3.3 After

```
0 BID 53.3000 x 8109 @ 8  ASK 53.3500 x 1936 @ 6
1 BID 53.2900 x 2482 @ 12 ASK 53.3600 x 2978 @ 7
2 BID 53.2850 x 1914 @ 4  ASK 53.3700 x 1988 @ 7
3 BID 53.2800 x 3829 @ 8  ASK 53.3800 x 814  @ 6
4 BID ******************* ASK *******************
```

2.3.4 MBLDeltaRefresh, action=BidRemoveLevelAndAppend

2.3.4.1 Before

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3100 x 2331 @ 10 ASK 53.3500 x 1936 @ 6
2 BID 53.3000 x 8109 @ 8  ASK 53.3600 x 2978 @ 7
3 BID 53.2900 x 2482 @ 12 ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.4.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:40:243,
              Market_utc_time=2010-11-08 16:22:40:243}
- Action=BidRemoveLevelAndAppend
- Level=1
- Price=53.28
- Qty={cumulated=2, nb_orders=2}
- ContinuationFlag=false
- OtherValues=()
```

2.3.4.3 After

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3000 x 8109 @ 8  ASK 53.3500 x 1936 @ 6
2 BID 53.2900 x 2482 @ 12 ASK 53.3600 x 2978 @ 7
3 BID 53.2800 x 2    @ 2  ASK 53.3800 x 814  @ 6
4 BID ******************* ASK *******************
```

2.3.5 MBLDeltaRefresh, action=BidClearFromLevel

2.3.5.1 Before

```
0 BID 53.3200 x 958  @ 3 ASK 53.3400 x 250  @ 1
1 BID 53.3100 x 1165 @ 5 ASK 53.3500 x 968  @ 2
2 BID 53.3000 x 4054 @ 4 ASK 53.3600 x 1489 @ 3
3 BID 53.2900 x 1241 @ 6 ASK 53.3700 x 994  @ 3
4 BID ****************** ASK *******************
```

2.3.5.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:41:159,
              Market_utc_time=2010-11-08 16:22:41:159}
- Action=BidClearFromLevel
- Level=2
- Price=<not relevant>
- Qty=<not relevant>
- ContinuationFlag=false
- OtherValues=()
```

2.3.5.3 After

```
0 BID 53.3200 x 958  @ 3 ASK 53.3400 x 250  @ 1
1 BID 53.3100 x 1165 @ 5 ASK 53.3500 x 968  @ 2
2 BID ****************** ASK 53.3600 x 1489 @ 3
3 BID ****************** ASK 53.3700 x 994  @ 3
4 BID ****************** ASK *******************
```

2.3.6 MBLDeltaRefresh, action=ALLClearFromLevel

2.3.6.1 Before

```
0 BID 53.3200 x 958  @ 3 ASK 53.3400 x 250  @ 1
1 BID 53.3100 x 1165 @ 5 ASK 53.3500 x 968  @ 2
2 BID ****************** ASK 53.3600 x 1489 @ 3
3 BID ****************** ASK *******************
```

2.3.6.2 Received Notification

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:22:48:158,
              Market_utc_time=2010-11-08 16:22:48:158}
- Action=ALLClearFromLevel
- Level=0
- Price=<not relevant>
- Qty={cumulated=<not relevant>, nb_orders=<not relevant>}
- ContinuationFlag=false
- OtherValues=()
```

2.3.6.3 After

```
0 BID ************************** ASK **************************
```

2.3.7 Continuation Flag

2.3.7.1 Initial Snapshot

```
0 BID 53.3200 x 1916 @ 6  ASK 53.3400 x 501  @ 3
1 BID 53.3100 x 2331 @ 10 ASK 53.3500 x 1936 @ 5
2 BID 53.3000 x 8109 @ 8  ASK 53.3600 x 2978 @ 7
3 BID 53.2900 x 2482 @ 12 ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.7.2 First Event

Layer 0 is locked on Price 53.34. The consistency of the cache book is not guaranteed with the current event.

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2013-12-03 16:49:11:147,
              Market_utc_time=2013-12-03 16:49:11:147}
- Action=BidInsertAtLevel
- Level=0
- Price=53.34
- Qty={cumulated=666, nb_orders=1}
- ContinuationFlag=true
- OtherValues=()
```

Bid and Ask sides are locked at price 53.34

2.3.7.3 Cache Book After First Event (ContinuationFlag=true)

```
0 BID 53.3400 x 666  @ 1  ASK 53.3400 x 501  @ 3
1 BID 53.3200 x 1916 @ 6  ASK 53.3500 x 1936 @ 5
2 BID 53.3100 x 2331 @ 10 ASK 53.3600 x 2978 @ 7
3 BID 53.3000 x 8109 @ 8  ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.7.4 Second Event

The book is then unlocked with a Delta that has the Continuation Flag set to False.

```
notification=MBLDeltaRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2013-12-03 16:49:11:147,
              Market_utc_time=2013-12-03 16:49:11:147}
- Action=AskRemoveLevel
- Level=0
- Price=0
- Qty={cumulated=0, nb_orders=<empty>}
- ContinuationFlag=false
- OtherValues=()
```

2.3.7.5 Cache Book After Second Event, Unlocked Cache Book (ContinuationFlag=false)

```
0 BID 53.3400 x 666  @ 1  ASK 53.3500 x 1936 @ 5
1 BID 53.3200 x 1916 @ 6  ASK 53.3600 x 2978 @ 7
2 BID 53.3100 x 2331 @ 10 ASK 53.3700 x 1988 @ 7
3 BID 53.3000 x 8109 @ 8  ASK ******************
4 BID ****************** ASK ******************
```

2.3.8 MBLOverlapRefresh

This section details:

- Fields' Description
- Overlap
- Overlap and Crop
- Overlap and Append.

2.3.8.1 Fields' Description

```
notification=MBLOvelapRefresh
- BidChangeIndicator=...
- AskChangeIndicator=...
- BidLimits=({ Price { QTY NbOrders } })
- AskLimits=({ Price { QTY NbOrders } })
```

For each side (bid, ask), a *list of order book entries* is provided along with an *indicator*. The list of entries can be empty. When not empty, the best prices display first. The indicator specifies:

- `start_level`—indicates the depth where you should place the corresponding order book entries
- `is_full`—a boolean value indicating if the order book entries span up to the bottom of the visible book.

At the `start_level`, you should copy the provided order book entries, overwriting any existing value (if any). When `is_full` is true, you should crop any previous entries that go deeper than the latest entry being updated.

To handle the indicator:

- in C++ API, use `FeedOS::Types::split OrderBookChangeIndicator (source_Indicator, is_full, source_start_level)`
- in C# API, use `MBLOverlapRefresh.splitOrderBookChangeIndicator`
- in Java API, interpret the indicator as shown the code below:

```
int start_level;
   boolean is_full;
   if (indicator < 0) {
      is_full=true;
      start_level=-indicator-1;
   } else {
      is_full=false;
      start_level=indicator;
   }
```

For more details, see also the following sections.

2.3.8.2 Overlap

2.3.8.2.1 Before

```
0 BID 13.3000 x 8122 @ 7  ASK 13.3400 x 51   @ 3
1 BID 13.2900 x 1112 @ 5  ASK 13.3500 x 936  @ 16
2 BID 13.2700 x 1996 @ 20 ASK 13.3600 x 8941 @ 5
3 BID 13.2600 x 718  @ 11 ASK 13.3700 x 985  @ 17
4 BID ******************* ASK ********************
```

2.3.8.2.2 Received Notification

`start_level=0` and `is_full=false`

```
notification=MBLOvelapRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:52:22:100,
              Market_utc_time=2010-11-08 16:52:22:100}
- BidChangeIndicator=0
- AskChangeIndicator=0
- BidLimits=( { 13.295 { 9885 10 } } { 13.28 { 3273 9 } } )
- AskLimits=()
- OtherValues=()
```

2.3.8.2.3 After

```
0 BID 13.2950 x 9885 @ 10 ASK 13.3400 x 51   @ 3
1 BID 13.2800 x 3273 @ 9  ASK 13.3500 x 936  @ 16
2 BID 13.2700 x 1996 @ 20 ASK 13.3600 x 8941 @ 5
3 BID 13.2600 x 718  @ 11 ASK 13.3700 x 985  @ 17
4 BID ******************* ASK ********************
```

2.3.8.3 Overlap and Crop

2.3.8.3.1 Before

```
0 BID 13.3000 x 8122 @ 7  ASK 13.3400 x 51   @ 3
1 BID 13.2900 x 1112 @ 5  ASK 13.3500 x 936  @ 16
2 BID 13.2700 x 1996 @ 20 ASK 13.3600 x 8941 @ 5
3 BID 13.2600 x 718  @ 11 ASK 13.3700 x 985  @ 17
4 BID ******************* ASK ********************
```

2.3.8.3.2 Received Notification

`start_level=1` and `is_full=true`

```
notification=MBLOvelapRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:52:22:100,
              Market_utc_time=2010-11-08 16:52:22:100}
- BidChangeIndicator=-2
- AskChangeIndicator=0
- BidLimits=( { 13.29 { 985 10 } } { 13.28 { 1173 19 } }  )
- AskLimits=()
- OtherValues=()
```

2.3.8.3.3 After

```
0 BID 13.3000 x 8122 @ 7  ASK 13.3400 x 51   @ 3
1 BID 13.2900 x 985  @ 10 ASK 13.3500 x 936  @ 16
2 BID 13.2800 x 1173 @ 19 ASK 13.3600 x 8941 @ 5
3 BID ******************* ASK 13.3700 x 985  @ 17
4 BID ******************* ASK ********************
```

2.3.8.4 Overlap and Append

2.3.8.4.1 Before

```
0 BID 13.3000 x 8122 @ 7  ASK 13.3400 x 51   @ 3
1 BID 13.2900 x 1112 @ 5  ASK 13.3500 x 936  @ 16
2 BID 13.2700 x 1996 @ 20 ASK 13.3600 x 8941 @ 5
3 BID ******************* ASK ********************
```

2.3.8.4.2 Received Notification

`start_level=0` and `is_full=false`

```
notification=MBLOvelapRefresh
- Code=<numeric instrument code>
- LayerID=0
- Timestamps={Server_utc_time=2010-11-08 16:52:22:100,
              Market_utc_time=2010-11-08 16:52:22:100}
- BidChangeIndicator=0
- AskChangeIndicator=0
- BidLimits=( { 13.31 { 985 10 } } { 13.30 { 1173 19 } } { 13.29 { 9185 2 } }
{ 13.28 { 233 3 } } )
- AskLimits=()
- OtherValues=()
```

2.3.8.4.3 After

```
0 BID 13.3100 x 985  @ 10 ASK 13.3400 x 51   @ 3
1 BID 13.3000 x 1173 @ 19 ASK 13.3500 x 936  @ 16
2 BID 13.2900 x 9185 @ 2  ASK 13.3600 x 8941 @ 5
3 BID 13.2800 x 233  @ 3  ASK ********************
4 BID ******************* ASK ********************
```

2.3.9 MBLMaxVisibleDepth

2.3.9.1 Before

```
0 BID 53.3100 x 2331 @ 10 ASK 53.3300 x 473  @ 1
1 BID 53.3000 x 8109 @ 8  ASK 53.3400 x 501  @ 3
2 BID 53.2700 x 1626 @ 2  ASK 53.3600 x 3273 @ 9
3 BID 53.2600 x 811  @ 5  ASK 53.3700 x 1988 @ 7
4 BID ******************* ASK *******************
```

2.3.9.2 Received Notification

```
notification=MBLMaxVisibleDepth
- Code=<numeric instrument code>
- LayerID=0
- MaxVisibleDepth=2
```

2.3.9.3 After

```
0 BID 53.3100 x 2331 @ 10 ASK 53.3300 x 473 @ 1
1 BID 53.3000 x 8109 @ 8  ASK 53.3400 x 501 @ 3
2 BID ******************* ASK ******************
```

2.4 Implementation Recommendations

If you do not want multilayer support, you should request only layer # 0 when calling:

- the `API QUOTATION SubscribeInstrumentsMBL` for the C++ API
- the `SubscribeMBL` for the C# API
- the `asyncQuotSubscribeInstrumentsMBL start` for the Java API.

If you want to implement multilayer support, you should start your subscription by requesting an empty list of layers. This means you are willing to process all available layers. In the callbacks `notif_MBLxxx()`, you should retrieve the layer ID and prepare to handle several layers.

On the derivatives markets:

- Layer 0 will be the “default order book” (either “full” or “outright-only”).
- Layer 1 (optional) will carry implied orders.

To easily maintain order books, we strongly recommend you use built-in API classes, such as:

- `InstrumentMBLData` for C++ API
- `MBLSnapshot` for C# API and Java API.

For each type of notification you receive (`FullRefresh`, `OverlapRefresh`, `DeltaRefresh`), you should call the corresponding method `updateXXXX()`.