Document title

EURONEXT CASH MARKETS - OPTIQ® KINEMATICS SPECIFICATIONS

Revision number	Date
1.3.0	31 Jan 2018

Number of pages

77

PREFACE

PURPOSE

The purpose of this document is to detail Kinematics for Optiq Order Entry Gateway and Market Data Gateway messages.

TARGET AUDIENCE

This document should be read by Euronext and Members using Optiq.

ASSOCIATED DOCUMENTS

The following list of the associated documents, which either should be read in conjunction with this document or which provide other relevant information for the user:

- Euronext Cash Markets – OEG Client Specifications – SBE Interface
- Euronext Cash Markets – OEG Client Specifications – FIX 5.0 Interface
- Euronext Cash Markets – CCG to OEG Changes Highlight
- Euronext Cash Markets – Optiq & TCS Error List
- Euronext Cash and Derivatives Markets – Optiq MDG Client Specifications
- Euronext Cash and Derivatives Markets – Optiq File Specifications

Clients are advised to also refer to the Euronext Rules and Regulations documents for more details.

For the latest version of documentation please visit http://www.euronext.com/optiq

SUPPORT

Optiq Support Desk

Tel: +33 1 70 48 25 55

Email: optiq@euronext.com

CONTENTS

1. OVERVIEW

1.1 INTRODUCTION

This document provides an overview of the exchange of messages between the Optiq Order Entry Gateway (OEG), the clients' systems and the Market Data Gateway (MDG). It includes:

- Typical trading scenarios and the corresponding public and private messages for these scenarios, and the different cases they may cover;
- The names and IDs of the messages sent;
- The events that trigger the transmission of messages.

This overview is meant to provide a description of the main structures and concepts used within this document, to facilitate the review of the individual topics and cases covered within.

The messages that are sent between trading members and Optiq are referred to as private messages whereas the messages that are sent by the external broadcasting systems are referred to as public messages.

Private messages are exchanged exclusively between the clients' system and the Optiq matching engine via order entry gateways, for example to request information from the system or to issue a command (e.g., enter an order). Private messages are also sent back by Optiq via order entry gateways (OEGs) to the client's system to provide the information requested, or confirm that a command has been successfully executed (or not), as well as to notify of trades, etc.

Public messages are sent by Optiq via MDG to provide to all subscribing clients with anonymized Market data, such as orders entered, best limits, trades executed, market events, updated limits, etc.

The diagrams in this document express representative examples of message sequences and other scenarios can be figured out from the ones described inside that document. The details of the message contents may vary depending on the example.

The diagrams also endeavour to represent as close to reality as possible the sequence in which events and steps occur, and messages are sent. This introduction provides indication when such cases are not feasible to represent faithfully due to complexity of graphical representation.

For a complete description of the messages and their fields, please refer to the associated document:

- Optiq Order Entry Gateway Messages Specifications SBE;
- Optiq Order Entry Gateway Messages Specifications FIX;
- Optiq Market Data Gateway Messages Specifications.

1.2   MESSAGE CODES AND NAMES

1.2.1   Private Messages

Possible Direction:
- Inbound - Client ▶OEG (From Client To OEG)
- Outbound - Client ◀OEG (To Client From OEG)

Order Entry Gateway message identifiers, which include message codes and names, are provided throughout the message kinematics section as shown below:

- For Inbound messages (example for **NewOrder** message):

    01 [D] NewOrder ⟶

      01 represents the SBE Bin Code.
      [D] represents the FIX Code.

- For Outbound messages (example for **Ack** message):

    ⟵ 03 [8] Ack

      03 represents the SBE Bin Code.
      [8] represents the FIX Code.

- When there is a difference of kinematics between SBE Bin and FIX protocols, the flows are distinguished as shown below (example of CollarBreachConfirmation) :

    20 CollarBreachConfirmation ⟶

    [G] OrderCancelReplaceRequest ⟶

      The SBE Bin message Code is represented alongside the FIX kinematic.

- The exhaustive list of SBE Bin and FIX message codes and names is provided in the table below:

SBE Bin Message Code	SBE Bin Message Name	FIX Message Code
01	New Order	D
03	Ack	8
04	Fill	8
05	Kill	8
06	Cancel Replace	G
07	Reject	9

SBE Bin Message Code	SBE Bin Message Name	FIX Message Code
08	Quotes	i
09	Quote Ack	b
10	Quote Request	AG
12	Cancel Request	F
13	Mass Cancel	q
14	Mass Cancel Ack	r
15	Open Order Request	AF
17	Ownership Request Ack	U29
18	Ownership Request	U18
19	Trade Bust Notification	8
20	Collar Breach Confirmation	
28	Price Input	UI
32	Liquidity Provider Command	UZ
33	Ask For Quote	UL
34	Request For Execution	UM
35	RFQ Notification	U35
36	RFQ Matching Status	U36
39	User Notification	CB
50	Instrument Synchronization List	U50
51	Synchronization Time	U51
100	Logon	A
101	Logon Ack	
102	Logon Reject	3
103	Logout	5
106	Heartbeat	0
107	TestRequest	1
108	TechnicalReject	

1.2.2   Public Messages

Possible Direction:
- Outbound - MDG ▶Client (From MDG To Client)

Market Data Gateway message identifiers, which include message codes and names, are provided throughout the message kinematics section as shown below:

For public messages sent to the Market:

1001   **MarketUpdate**

■   The exhaustive list of message codes and names is provided in the table below:

Message Code	Message Name
1001	Market Update
1002	Order Update
1003	Price Update
1004	Full Trade Information
1005	Market Status Change
1006	Timetable
1007	Standing Data
1008	Real Time Index
1009	Statistics
1011	Index Summary
1012	Strategy Standing Data
1013	Contract Standing Data
1014	Outright Standing Data
1101	Start Of Day
1102	End Of Day
1103	Health Status
1104	Trade Retransmission Start
1105	Trade Retransmission End
2101	Start Of Snapshot
2102	End Of Snapshot

1.2.3   Graphical representations

The diagrams in this document represent the following components:

■   The overall Optiq system which is the new integrated trading platform for the Euronext markets, shown as below:

■   The Order Entry Gateway which is the private interface between clients and the matching engine:

                                       1.3.0

■   The Market Data Gateway (MDG) which sends public messages to the Market:

MDG

■   The clients' systems, used by the client to send and receive private messages to and from the matching engine, here referred to as Broker:

Broker

■   And the Market represents all the publicly available data sent by the exchange to all subscribers of the public feeds:

Market

Note: for readability purposes the field names in the graphs are abbreviated, e.g. *Order Quantity* is referred to as *OrderQty*, etc.

1.2.4   Main Principles

A request sent by a client will usually:

■   Trigger an outbound acknowledgment message from the matching engine which is exclusively sent to this client, and in some cases this can be followed by other notification messages;

■   Trigger one or several market data messages if the request has a direct impact on the Central Order Book (COB).

Below is an abbreviated, generic example of the interaction of messages, for the submission of a **NewOrder** (01) (FIX D) message:
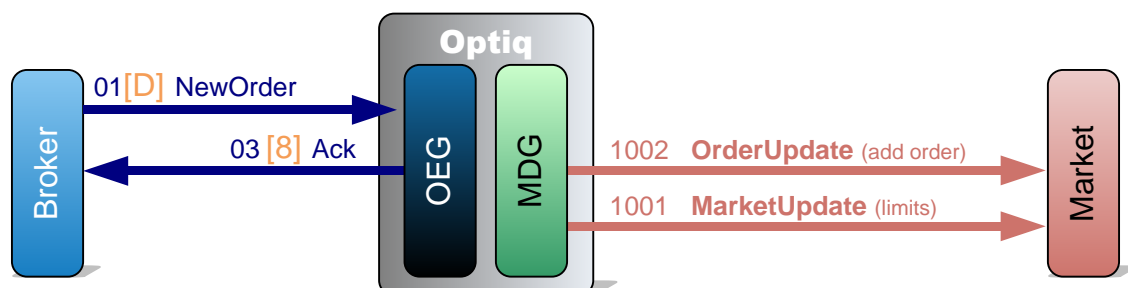
When required diagrams may include division into steps of the scenarios displayed, that are delineated by dotted lines, and are denoted by the number of the step. Numbers denoting the steps in the diagram correspond to the numbers used in the explanation below the diagram.

More detailed diagrams may include additional details for the individual messages, such as, Side, Order Priority, Price, Quantity, etc.

1.2.5   Important Notes

1.2.5.1   Private and Public feed reconciliation

The following diagram explains how the members can reconcile their orders across the Private and Public data feed using the field named *Order Priority*. Please review the note on *Order Priority* field at the end of this section.
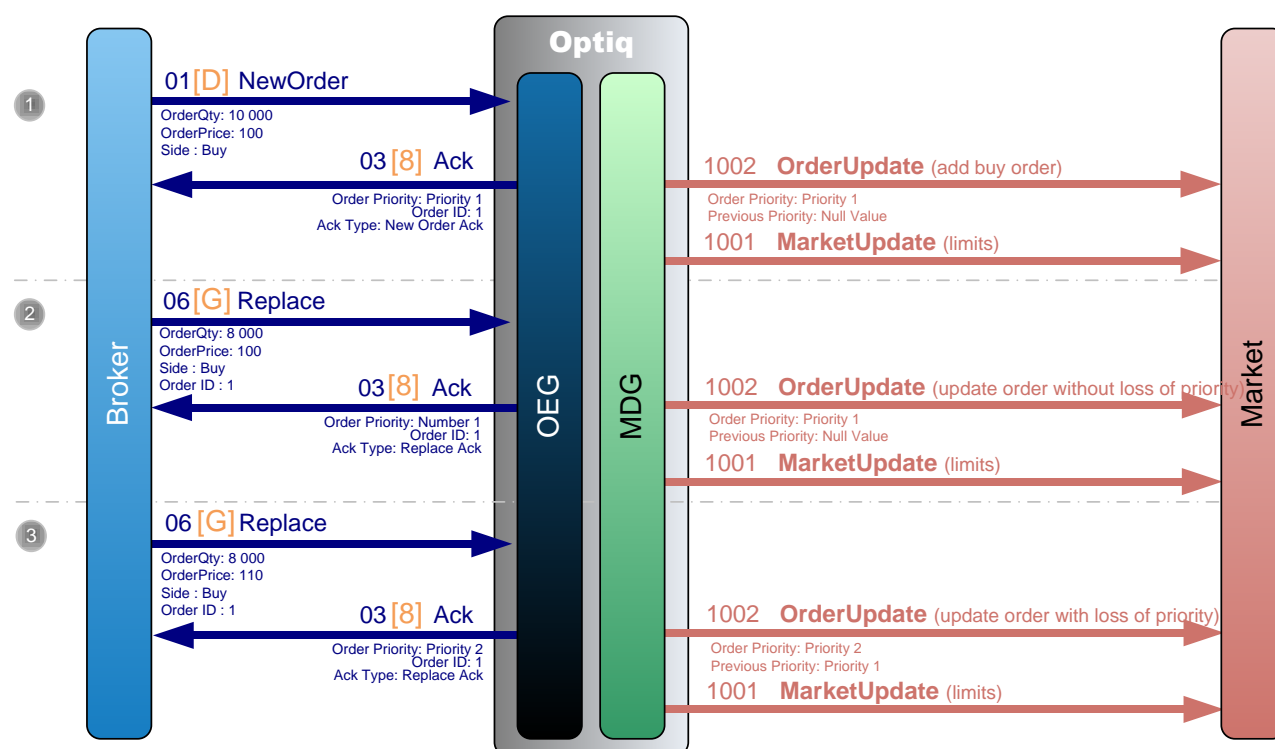
① A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new buy order with a quantity of 10,000 and a price at 100.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order. This message provides the value of the *Order Priority* that is used in the market data feed as the order identifier.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market, by MDG, to add the order identified by its *Order Priority*, followed by another **MarketUpdate** (1001) message to update the limits.

②   The same Broker sends a private **Replace** (06) (FIX G) (message to reduce the quantity of its order to 8,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order modification.

A public **OrderUpdate** (1002) message is sent to the market, by MDG, to update the order quantity, identified by its *Order Priority*, and another public **MarketUpdate** (1001) message to update the limits.

③   Then the same Broker sends another private **Replace** (06) (FIX G) message to change the price of its order to 110.

OEG sends back a private **Ack** (03) (FIX 8) message the successful receipt and technical processing of the order modification.

A public **OrderUpdate** (1002) message is sent to the market to update the order price, identified by its *Previous Priority* and its new *Order Priority*, and another public **MarketUpdate** (1001) message to update the limits.

Note: *Order Priority* is a numeric value, representing the priority of the order. The order with the lowest value of *Order Priority* has the highest priority. *Order Priority* is unique per Symbol Index and EMM.

Order Priority is used as the order unique identifier for the market data feed. *Previous Priority* is populated only when there is an update with loss of priority and for deletion of orders.

Hence in case of update with loss of priority, clients should remove from their market sheet the order identified in the *Previous Priority* and add a new order with the *Order Priority* newly provided.

1.2.5.2   Trade, Collars and Full Trade Information generation

A public message **FullTradeInformation** (1004) is sent in a dedicated Post-Trade channel each time a **MarketUpdate** (1001) following a trade is disseminated to the market by MDG. But for readability purposes it is not shown on the kinematics diagrams.

And every time there is change in the reference price, dynamic collars are disseminated in the **MarketUpdate** (1001) message with the trade. But for readability purposes it is not shown on the kinematics diagrams.

Example:

The Last Traded Price of an instrument was 100, low dynamic collar was 95, and high dynamic collar was 105.

If a trade occurs and modifies the LTP to 101 (low dynamic collar is updated to 96, and high dynamic collar is updated to 106) then the following sequence of public messages will be sent by MDG:

1001   **MarketUpdate (Trade + Collars)**

1004   **FullTradeInformation**

However, to improve readability, in cases whenever trades occur and cause an update of the dynamic collars only the following message will be represented in the diagrams:

1001   **MarketUpdate (Trade)**

1.2.5.3   Simultaneity of Private Messages

In all the diagrams of this document multiple private messages resulting from the same event (eg. **Fill** (04) (FIX 8) messages due to a trade execution) are represented as if they were sent one by one:
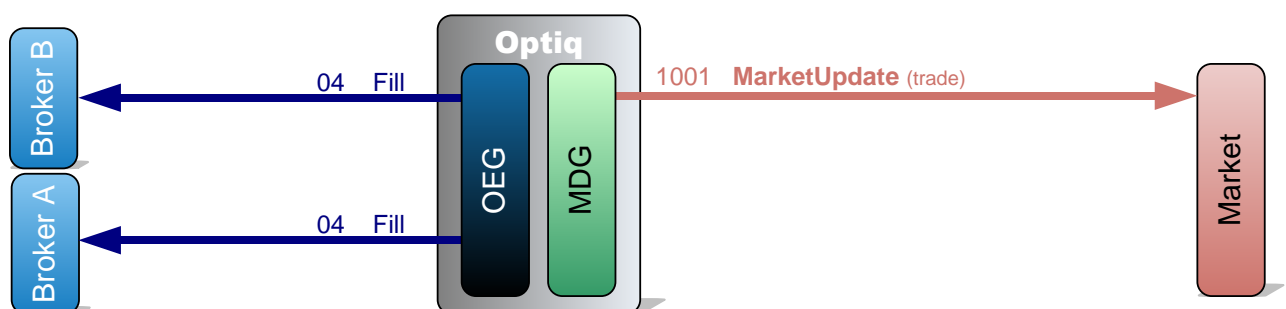
This is done to reduce complexity of the graphical representation and to improve readability. In reality such messages are sent at the same moment to the different brokers:
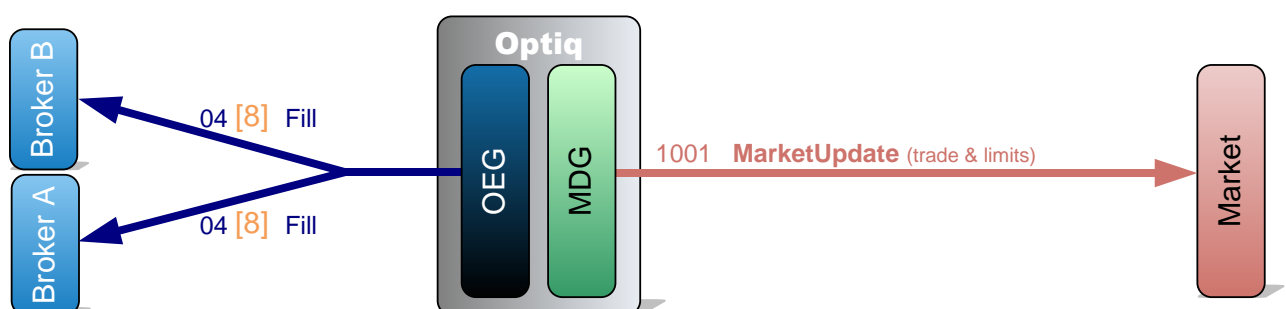
For the rest of this document please assume that messages resulting from the same event and sent to different clients are sent at the same moment.

1.2.5.4   Market Update and Order Update generation

All kinematics described in this document are common across Regulated Markets and Warrants Markets unless specified in brackets (as it is the case for Warrants kinematics). The following conventions have been adopted:

■ For all Cash Equities, Bonds, and ETF Regulated Markets (excluding Best Of Book [BOB], as well as the Warrant & Certificates on the new Market Models) Optiq MDG will provide **OrderUpdate** messages and **MarketUpdate** messages (containing only BBOs and not the depth of the order book).

The **MarketUpdate** message always follows an **OrderUpdate** message when notifying a limit (BBO). In case of a trade a **MarketUpdate** message is disseminated for the trade, followed by an **OrderUpdate** message to update the book and another **MarketUpdate** message for the update of the limits (BBO).

■ For BOB available within the Cash Markets, Optiq MDG will **not** provide **OrderUpdate** messages at all: it will only provide **MarketUpdate** messages, with the full depth of the order book (no BBO).

■ For the Warrant & Certificates on the new Market Model, Optiq MDG will **not** provide **OrderUpdate** messages at all: it will only provide **MarketUpdate** messages for BBOs and for the full depth of the order book.

Example:

After a trade takes place, the following MDG messages are displayed in the kinematics diagrams:

1001   **MarketUpdate** (Trade) ⟶

1002   **OrderUpdate** (Update Buy and/or Sell orders) ⟶

1001   **MarketUpdate** (Limits) ⟶

This scenario is different for the kinematics of the Regulated Market and Warrant & Certificates on the New Market Models:

For Regulated Market (except BOB), it should be read as followed:

1001   **MarketUpdate** (Trade) ⟶

1002   **OrderUpdate** ⟶
(Delete/Update Order X + Add New/Update Order Y)

1001   **MarketUpdate** (BBOs) ⟶

Remark: There is **no** dissemination of the **market depth** in the MDG **MarketUpdate** (1001) message.

For BOB, available within the Cash Markets, the scenario is as follows:

1001   **MarketUpdate** (Trade) ⟶

1001   **MarketUpdate** (Limits) ⟶
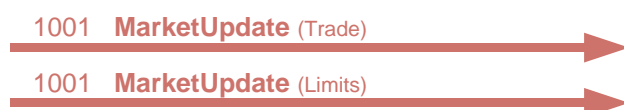
Remark: There is **no** MDG **OrderUpdate** (1002) message dissemination for BOB.

For the Warrant & Certificates on the New Market Model, the scenario is as follows:

1001   **MarketUpdate** (Trade)

1001   **MarketUpdate** (Limits + BBOs)

Remark: There is **no** MDG **OrderUpdate** (1002) message dissemination for Warrant & Certificates on the New Market Model.

2.   COMMON KINEMATICS

The following public messages contain repeating sections:

- ▪ PriceUpdate;
- ▪ OrderUpdate;
- ▪ MarketUpdate;
- ▪ MarketStatusChange.

Detailed information regarding repeating sections can be found in the documents **Euronext Cash and Derivatives Markets – MDG Client Specifications**.

2.1   TRADING SESSION MANAGEMENT

2.1.1   Initialisation of a New Trading Day
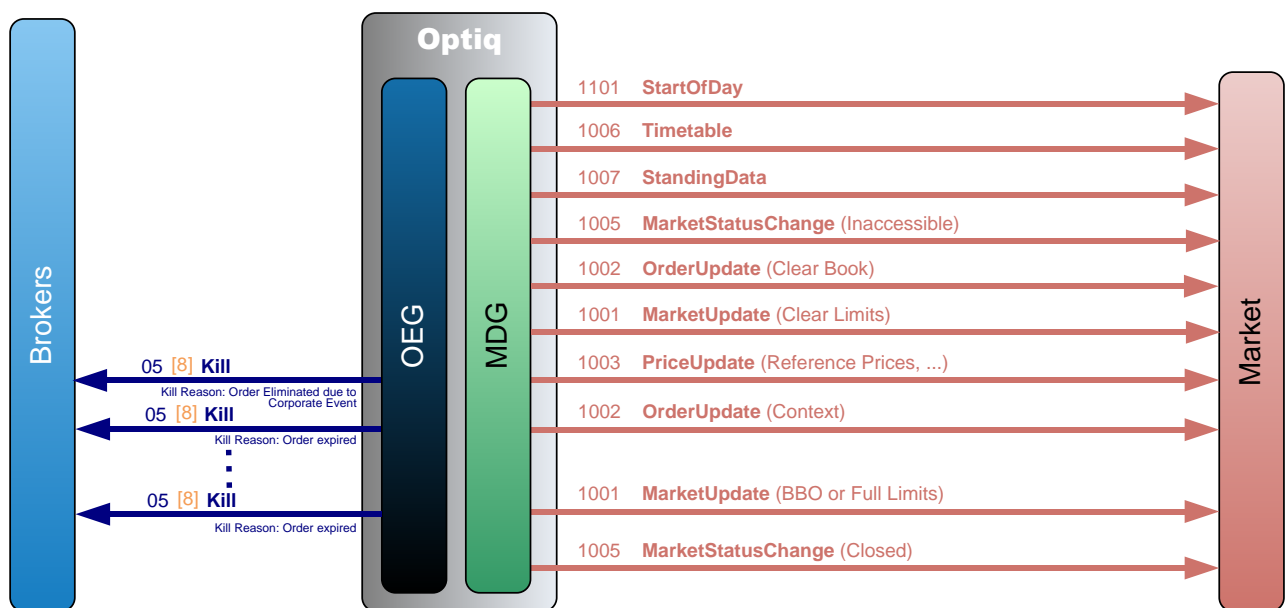
At the initialization of each new trading day the Exchange sends the following public messages (the generation sequence is guaranteed to always be the same):

- **StartOfDay** (1101) message: It is always the first message of the day, which indicates the date of the trading session.

- **Timetable** (1006) message: It provides all the trading patterns that are used across all the instruments.

- **StandingData** (1007) message: For every single instrument it provides to the members all the necessary data for the trading day.

- **MarketStatusChange** (1005) message: For every single instrument it is sent with *Book State* set to 'Inaccessible', *Trading Period* set to 'Opening' and *Rebroadcast Indicator* set to '0'.

　　　　　　　　　　1.3.0

- **OrderUpdate** (1002) message: For every single instrument on order driven market, the order book is cleared at the beginning of the day.

- **MarketUpdate** (1001) message: For every single instrument the limits are cleared at the beginning of the day.

- **PriceUpdate** (1003) message: For every single instrument, it provides all updated reference prices complementary to the BBO for trading (Closing Prices, Uncrossing Price, Valuation Prices, Min/Max Out of Session Trade Prices, Net Asset value for eligible instruments).

- **OrderUpdate** (1002) message: All the live orders are broadcasted at the beginning of each trading day for each eligible instrument, with *Rebroadcast Indicator* set to '1'. All the expired orders (GTD and GTC orders reaching expiry date) are removed from the scope of lives orders and not sent to the Market. Thus a private **Kill** (05) (FIX 8) message will be sent for each expired order.

- **MarketUpdate** (1001) message: For every single instrument it provides either the BBO for order-driven markets or all the limits for the other markets, with *Rebroadcast Indicator* set to '1'.
  If relevant, static and dynamic collars are also sent after the limits.

- **MarketStatusChange** (1005) message: For every single instrument a *Book State* set to 'Closed' is sent at the scheduled time.

On private messaging side, Changes due to previous trading day Corporate Events triggering cancellation of orders will be communicated as Kill (05) (FIX 8) messages at the start of day.

2.1.2　End Of Day
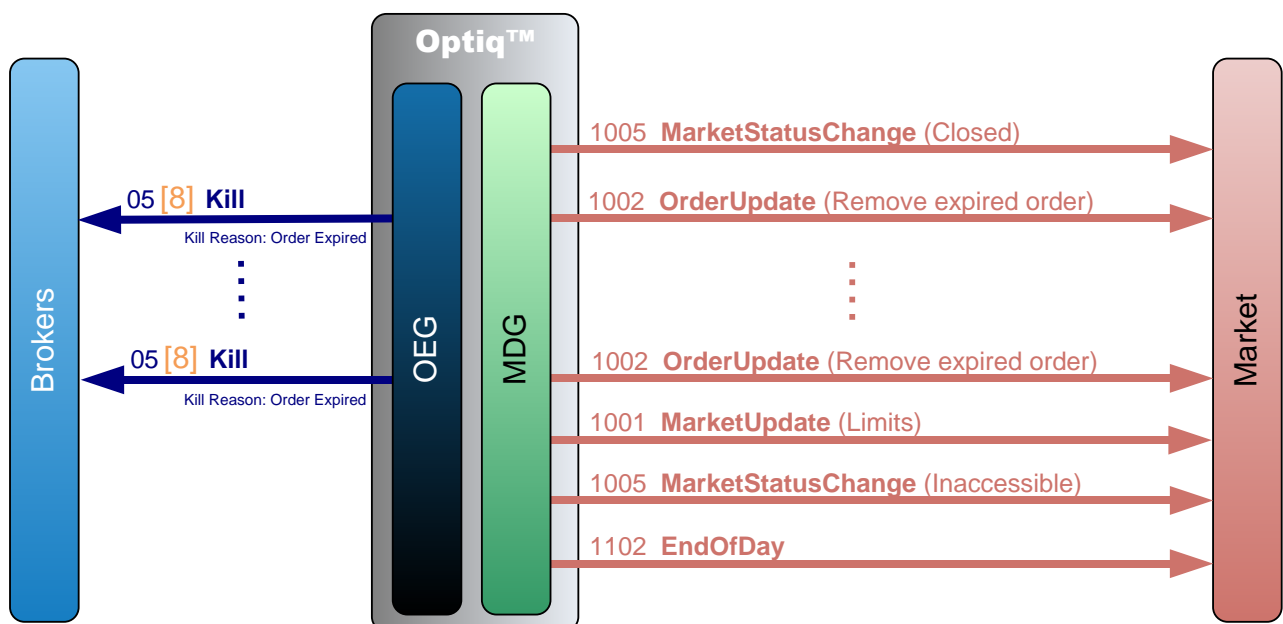
At the end of the trading day, when the instrument is in Closed State, expired orders (Day orders only) are killed, thus a private **Kill** (05) (FIX 8) message along with a public **OrderUpdate** (1002) message will be sent for each expired order, along with **MarketUpdate** (1001) for the limits.

At the scheduled time a **MarketStatusChange** (1005) message is sent for the Inaccessible phase.

The public **EndOfDay** (1102) message is always the last message sent by the Exchange, it notifies that the platform and its network are now closed (members cannot send nor receive messages until the next trading day).

Note: Clients should be aware that orders eliminated at the end of the session will not be re-broadcast at the start of the next trading session. In case of disconnection at the end of the sessions, clients are advised to remove any expired Day orders from their book.

2.2   ADMIN MESSAGES

Please note that all administrative messages exchanged between the client and the exchange are issued per OE session (physical connection).

2.2.1   Successful Logon
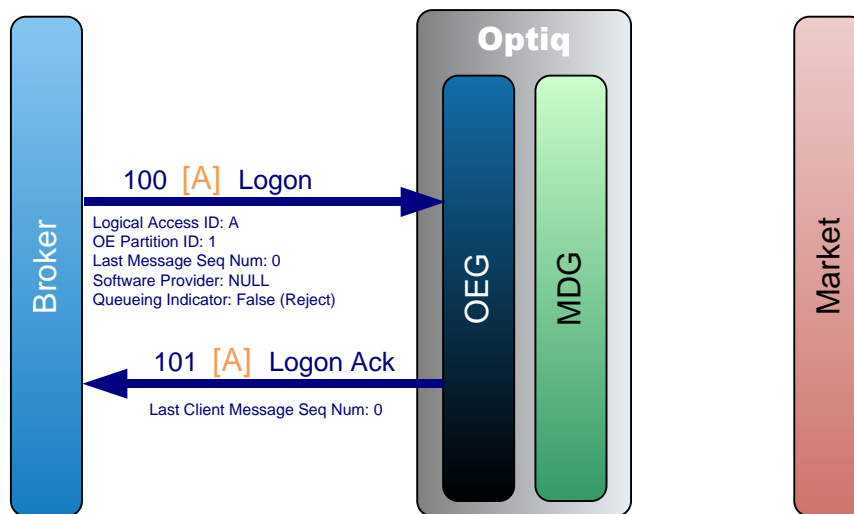
At the beginning of each trading day the members must log on to the Order Entry Gateway prior to send any other message.

In order to initiate the connection the broker sends a **Logon** (100) (FIX A) message. If the logon is successful the OEG sends back a **LogonAck** (101) message (only in SBE Bin protocol).

In FIX protocol, if the logon is successful the OEG sends back a **Logon** (A) message. While in SBE the sequence numbers start from 0, in FIX the sequence numbers start from 1.

No message is sent to the Market.

2.2.2   Logon Rejection

Logon rejection behaviour prescribed by the FIX protocol is different from that adopted for SBE, and for this case two different diagrams are provided, each one specific to the protocol. While at high level the behaviour might be different, the result of the Logon Rejection will be the same.

2.2.2.1   Logon Rejection in SBE
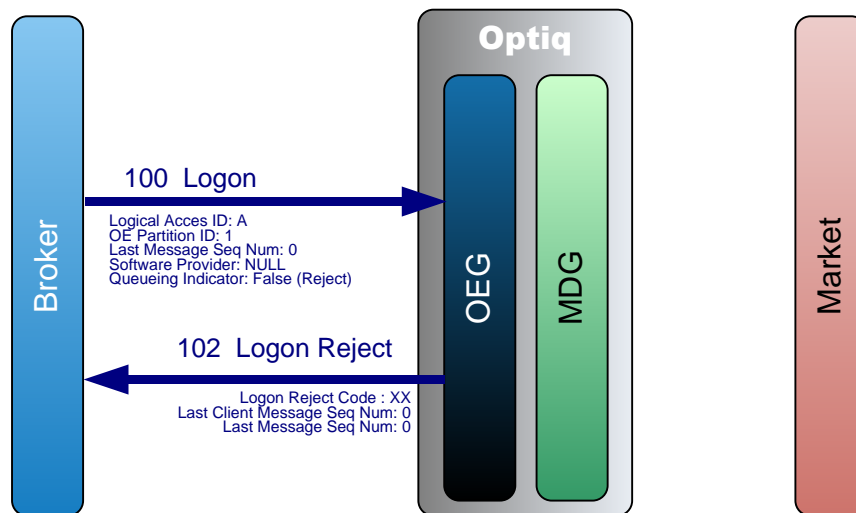
A broker sends a **Logon** (100) message in order to initiate the connection with the OEG. If for any reason the **Logon** (100) message is not accepted, the OEG sends back a **LogonReject** (102) message. No message is sent to the Market.
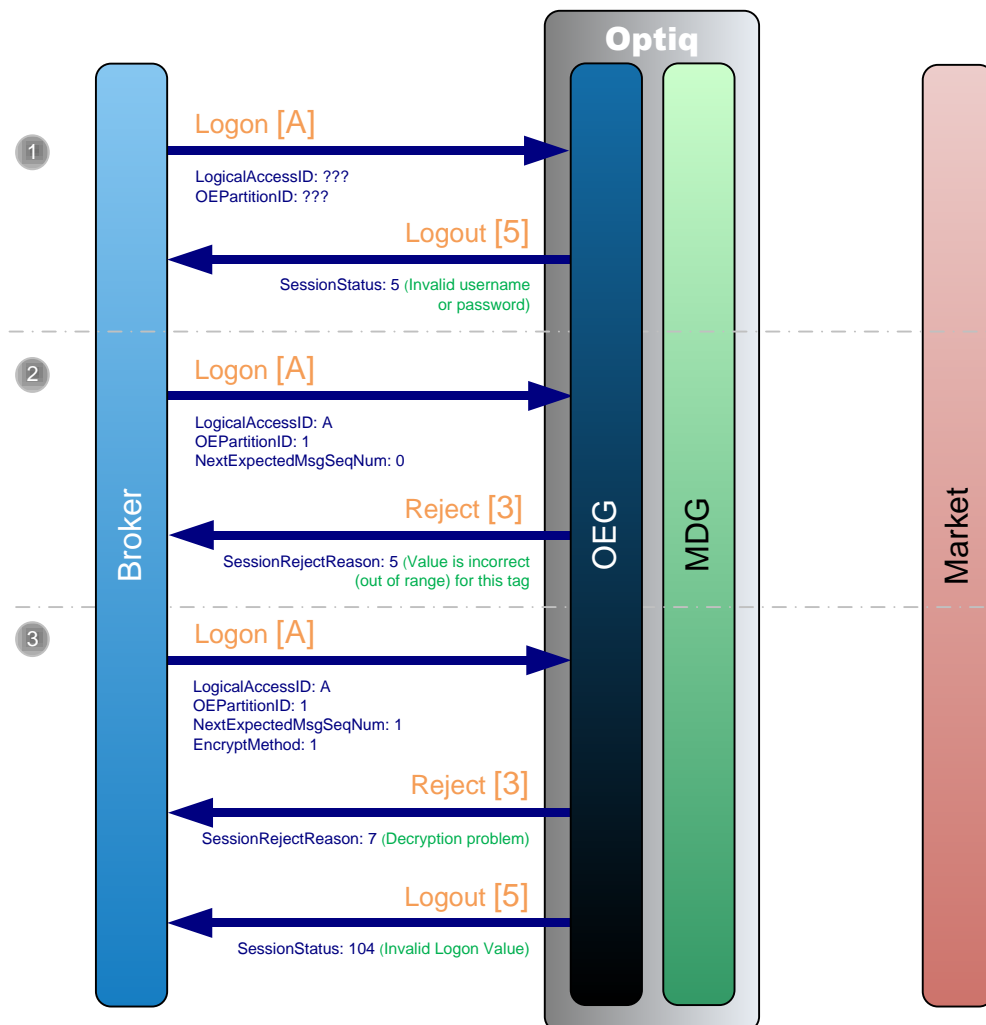
19 of 77   1.3.0

2.2.2.2   Logon Rejection in FIX

A broker sends a **Logon** (A) message in order to initiate the connection with the OEG. If for any reason the **Logon** (A) message is not accepted, the OEG sends back a **Logout** (5) message.

Additionally, OEG sends a **Reject** (3) message if the **Logon** (A) is poorly formatted.

①   A broker sends a **Logon** (A) message in order to initiate the connection with the OEG. If the fields *LogicalAccessID* and *OEPartitionID* are wrong or not recognized for the associated *SenderCompID*, OEG sends back a **Logout** (5) message with *SessionStatus* set to '5' (Invalid username or password).

②   A broker sends a **Logon** (A) message in order to initiate the connection with the OEG. If the field *NextExpectedMsqSeqNum* is set to '0', OEG sends back a **Reject** (3) message with SessionRejectReason set to '5' (Value is incorrect (out of range) for this tag).

③   A broker sends a **Logon** (A) message in order to initiate the connection with the OEG. If the value of the field *EncryptMethod* is different than '0', OEG sends back a **Reject** (3) message with SessionRejectReason set to '5' (Value is incorrect (out of range) for this tag).

No message is sent to the Market.

2.2.3   Logout

Logout behaviour prescribed by the FIX protocol is different from that adopted for SBE, and for this case two different diagrams and descriptions of steps are provided, each one specific to the protocol. While at high level the behaviour might be different, the result of the Logout from the system will be the same.

Logout is used to improve session management processes. This message identifies to the exchange if the client has disconnected on purpose or because of technical issue.

Note: This will trigger the Cancel on Disconnect mechanism (only on the specific orders on which it is enabled).

2.2.3.1   Logout in SBE
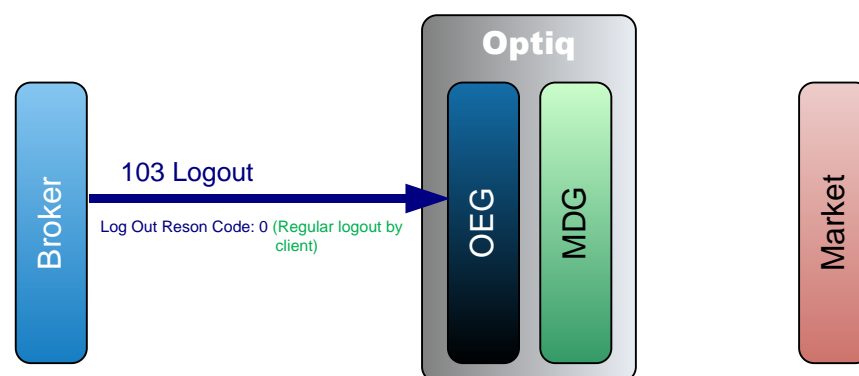
In order to log out the broker sends a **Logout** (103) message, OEG immediately closes the physical connection. No message is sent to the Market.

2.2.3.2   Logout in FIX
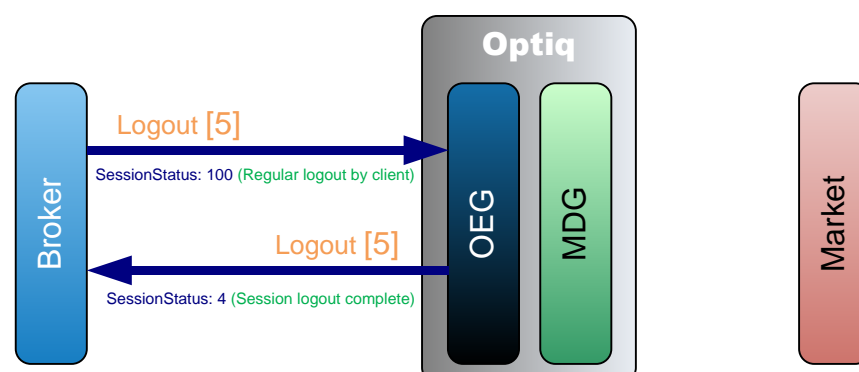
In order to log out the broker sends a **Logout** (5) message with *SessionStatus* set to '100' (Regular logout by client). In response OEG firstly sends back a **Logout** (5) confirmation message with *SessionStatus* set to '4' (Regular logout complete) and then closes the physical connection.

No message is sent to the Market.

2.2.4 Heartbeat
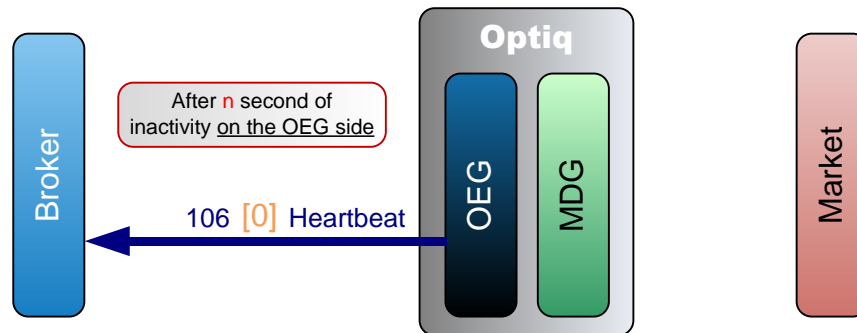
After *n* second(s) of inactivity on its side (ie. when the OEG has not sent any message since *n* second(s)) the OEG sends a **Heartbeat** (106) (FIX 0) message to the broker. The broker does not have to respond anything; it is only a notification from the OEG.

Note: The value of *n* will be provided for each Optiq Segment in the *Connectivity Specifications*.

2.2.5 Test Request
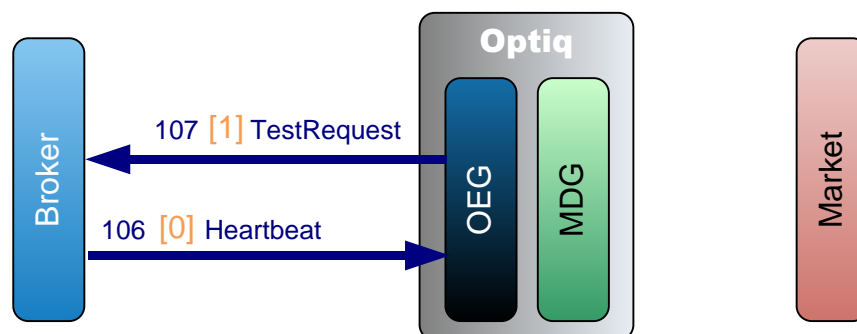
After *n* second(s) of inactivity on the broker side (i.e. when the OEG has not received any message since *n* second(s)) the OEG sends a **TestRequest** (107) (FIX 1) message to the broker.

- If the broker issues a message in the following *n* second(s), the **TestRequest** (107) (FIX 1) is ignored. Note the message issued by the member can be an **Heartbeat** (106) (FIX 0) message or any other application message (such as **NewOrder** (01) (FIX D), **CancelReplace** (06) (FIX G)…).
- If the broker does not issue any message in the following *n* second(s), the OEG closes the connection. (It triggers the Cancel on Disconnect mechanism on the specific orders on which it is enabled).

The **TestRequest** (107) (FIX 1) message can also be sent by the Broker, in this case the OEG will respond with a **Heartbeat** (106) (FIX 0) message:
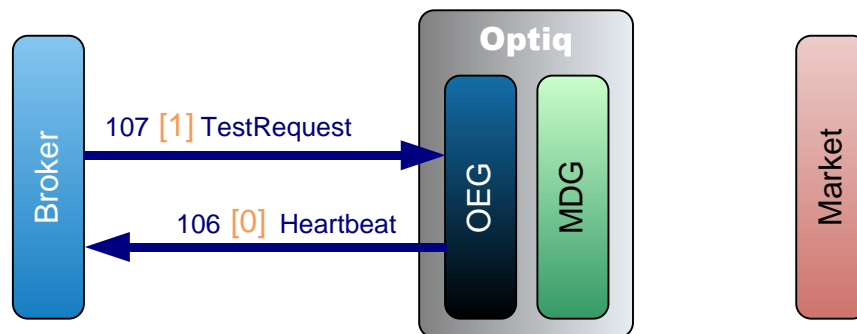
2.3   ENTERING AN ORDER

2.3.1   Incoming Order Fully Matched

①   A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 10,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

② Another Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 8,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to publish the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for the trade, along with a public **OrderUpdate** (1002) message to update the first order (Buy) with remaining quantity (2 000), followed by another **MarketUpdate** (1001) message to update the limits.

Note: There is no removal of the sell order in the last public **OrderUpdate** (1002) message as it is immediately matched and thus never enters the book.

2.3.2 Incoming Order Partially Matched

① A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 8,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

② Another Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 10,000.
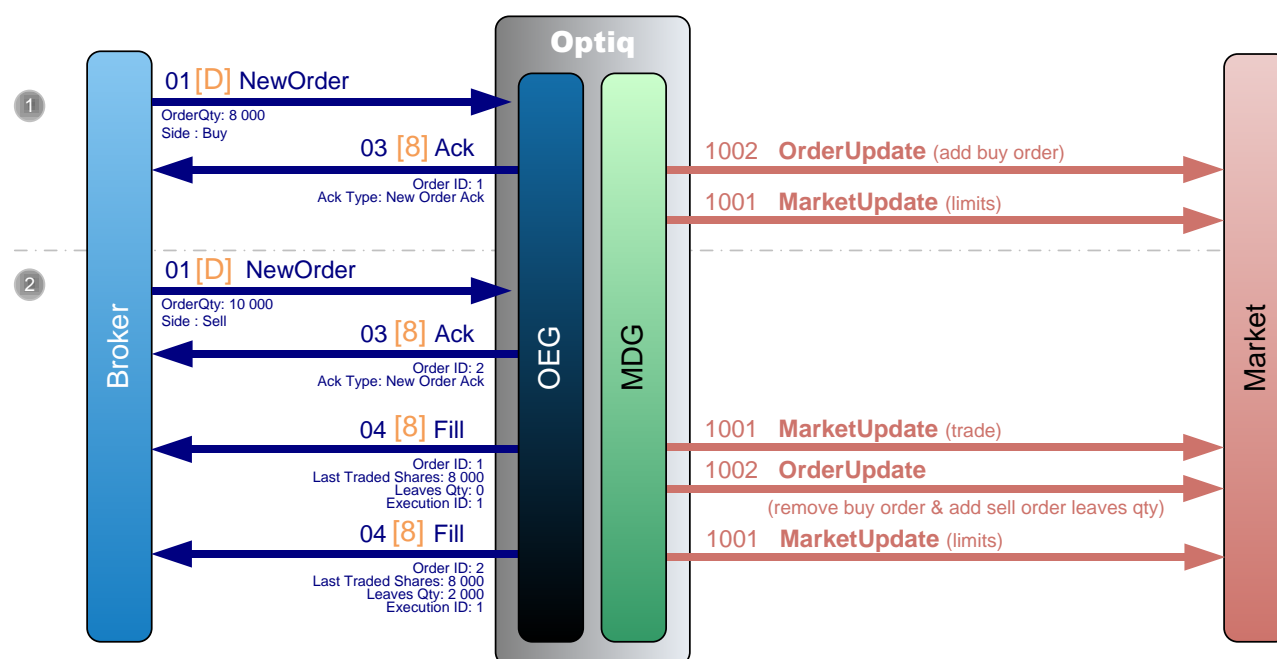
OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for the trade along with a public **OrderUpdate** (1002) message to remove the Buy order and add the Sell order with its leaves quantity (2 000), followed by another **MarketUpdate** (1001) message to update the limits.

2.3.3   Cross Order (New Order with a Cross side)

① A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new order with a quantity of 10,000 and a Cross side.

OEG sends back two private **Ack** (03) (FIX 8) messages to confirm the successful receipt and technical processing of the cross order.

The entering Cross order is immediately filled for its total quantity of 10,000 and OEG sends back two private **Fill** (04) (FIX 8) messages to the broker to notify the trade full execution.

A public **MarketUpdate** (1001) message is sent to the market for the trade.

2.3.4   New Order Rejected

①   A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new buy order with a quantity of 10,000.

If the order is rejected OEG sends back a private **Reject** (07) (FIX 8) message with an Error Code. The reason of the rejection can be found using the Error Code within the *Error list document*. No message is sent to the Market.

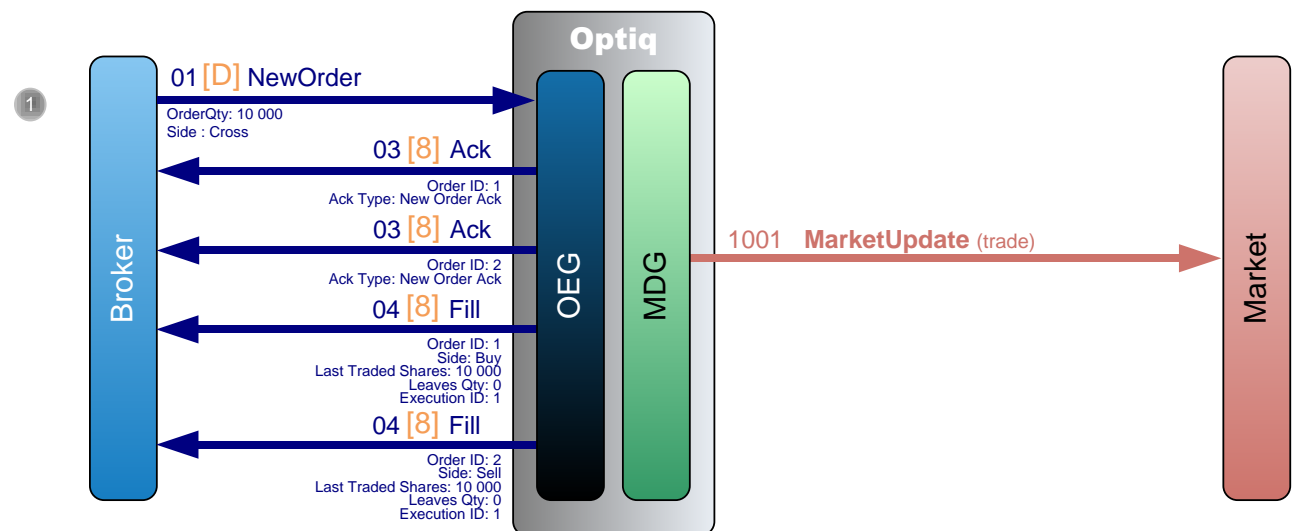### 2.3.5   Immediate Or Cancel Order Partially Filled
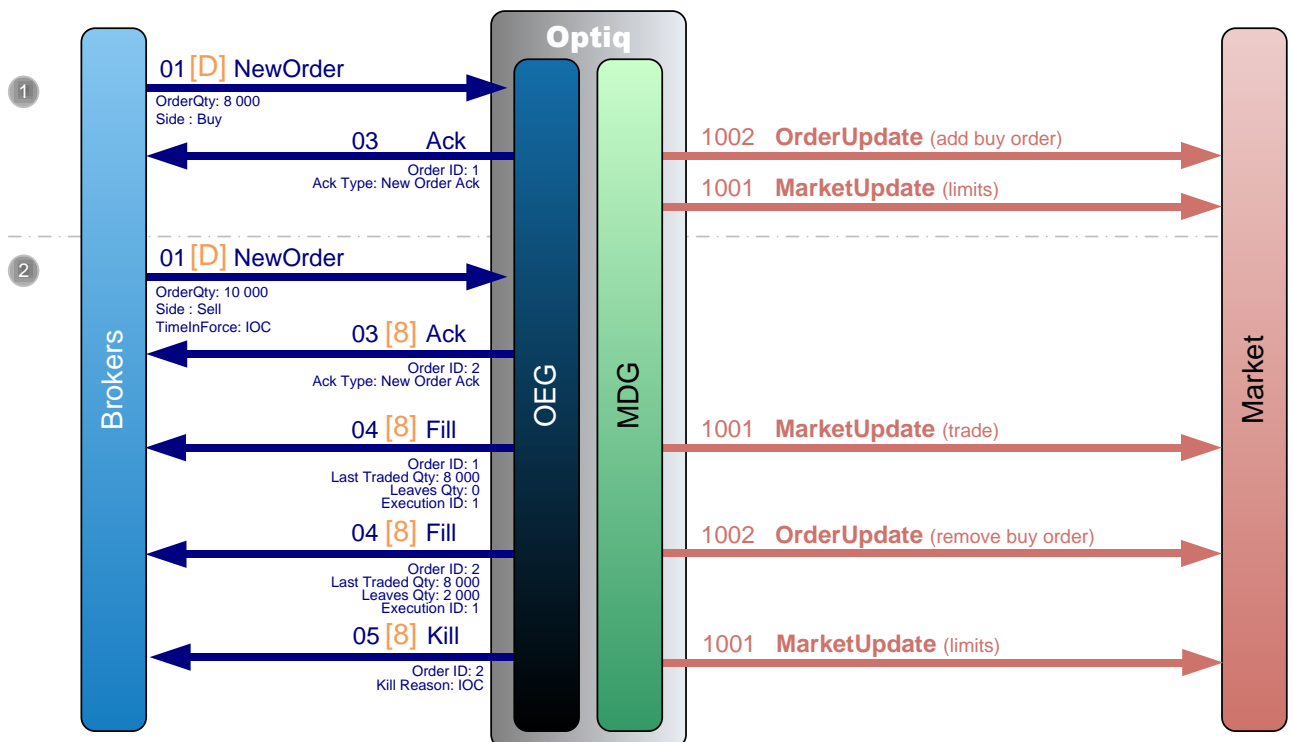
①   A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 8,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limit.
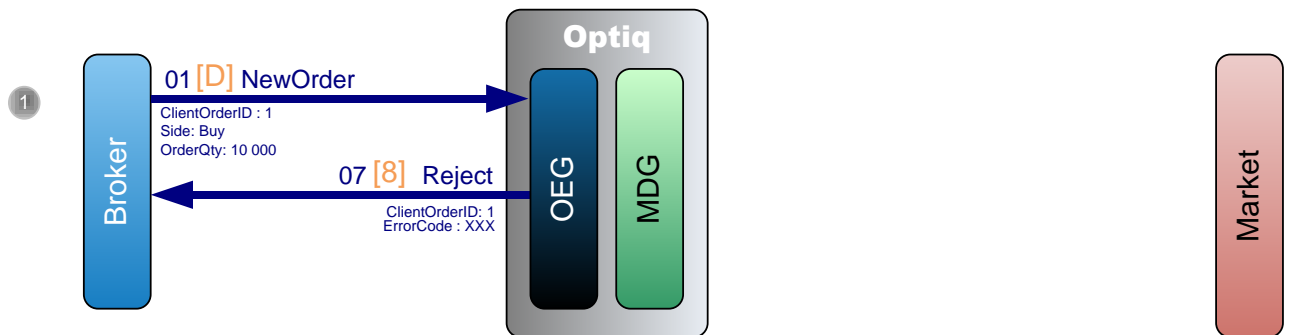
② Another Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 10,000 and a validity condition of Immediate or Cancel (IOC).

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The entering order immediately matches the first order for a quantity of 8,000 and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution. As the remaining quantity cannot be immediately filled, OEG sends back to the second Broker a **Kill** (05) (FIX 8) message to cancel it.

A public **MarketUpdate** (1001) message is sent to the market for the trade along with a public **OrderUpdate** (1002) message to remove the first order (Buy), followed by another **MarketUpdate** (1001) message to update the limits.

2.3.6   Market to Limit Order Partially Filled
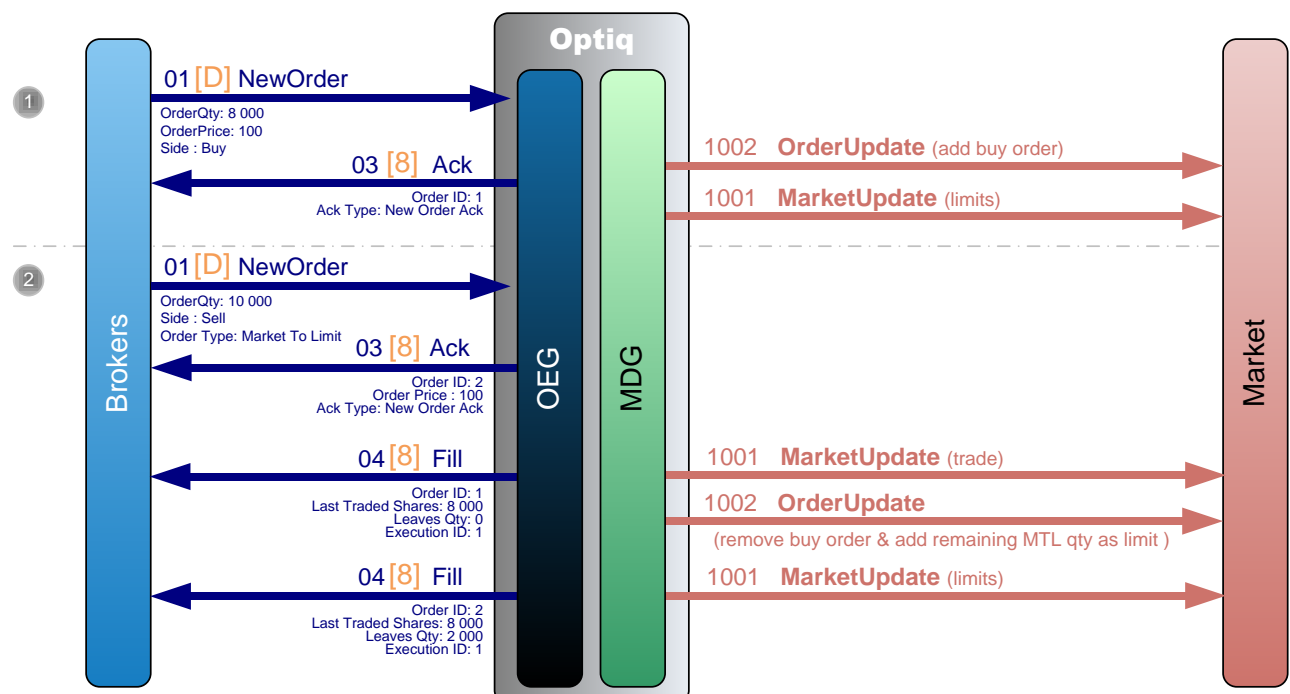
① A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 8,000 a price of 100.

OEG sends back a private **Ack (03) (FIX 8)** to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

②   Another Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 10,000 and a Market To Limit (MTL) order type.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order, with *Ack Type*=New Order Ack, *Order Price*=100.

The entering order immediately matches the first order for a quantity of 8,000 and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for the trade along with a public **OrderUpdate** (1002) message to remove the first order (Buy) and add the remaining quantity of the MTL (sell) order transformed into a Limit order at the price of 100, followed by another **MarketUpdate** (1001) message to update the limits.

2.3.7 Triggered Stop Orders
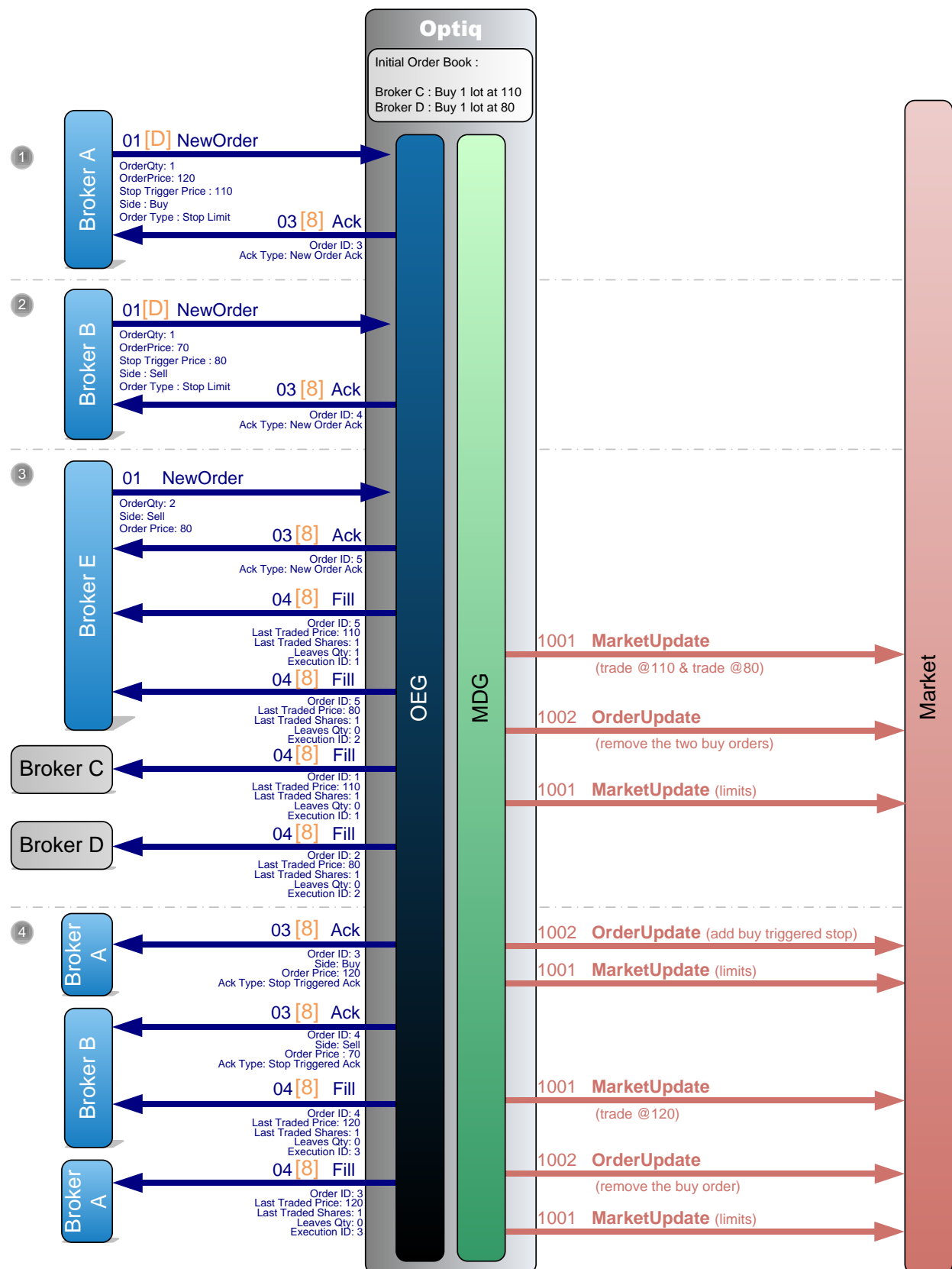
There are already two limit orders in the order book, one buy at price=110 and quantity=1 and another buy at price=80 and quantity=1.

① **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Stop Limit Buy order with Trigger=110 and Price=120.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

No public message is generated.

② **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Stop Limit Sell order with Trigger=80 and Price=70.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

No public message is generated.

③ **Broker E** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell Limit order with a quantity of 2 and a price of 80.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

This order matches the two Buy orders that are in the order book (price=110 and price=80), so OEG generates four private **Fill** (04) (FIX 8) messages, two for the Buy orders and two for the Sell order. And this triggers the two Stop Limit Orders.

A public **MarketUpdate** (1001) message is sent to the market for the two trades (at 110 and 80) and the updated limits, along with a public **OrderUpdate** (1002) message to remove the first two orders (Buy).

④ OEG sends a private **Ack** (03) (FIX 8) message to **Broker A** to confirm the trigger of the Stop Limit Buy order.

The triggered Stop Limit Buy order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the buy order (price=120).

OEG sends a private **Ack** (03) (FIX 8) message to **Broker B** to confirm the trigger of the Stop Limit Sell order.

The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker ('A' & 'B') to notify the trade execution at price=120.

A public **MarketUpdate** (1001) message is sent to the market for the trade (at 120) and the updated limits, along with a public **OrderUpdate** (1002) message to remove the first order (Buy).

Note: The first Ack (for the new Stop order) and the second Ack (for the triggered Stop order) are differentiated by the *Ack Type*.

                                       1.3.0

2.3.8   Iceberg Order Refilled

①   **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy Iceberg order with a quantity of 1,000 and a Disclosed Quantity of 300.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order for 300 lots and a public **MarketUpdate** (1001) message to update the limit.

②   **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 300.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution for a quantity of 300.

A public **MarketUpdate** (1001) message is sent to the market for the trade of 300 lots along with a public **OrderUpdate** (1002) message to remove the first order (Buy) , followed by another **MarketUpdate** (1001) message to update the limits.

③   Then OEG sends a private **Ack** (03) (FIX 8) message to **Broker A** to notify the refill to the broker (Refilled Iceberg Ack) and provides the *Order Priority* of the order which allows to reconcile the order in private and public messages.

A public **OrderUpdate** (1002) message is sent to the market to add a new buy order of 300 lots along with a public **MarketUpdate** (1001) message to update the limit.

Note: In the case of the randomisation for the displayed quantity of the Iceberg order, the member must reconcile its order in the market data feed (by using the *Order Priority*) to know exactly the disclosed quantity of the order. Please note that the randomly refilled disclosed quantity can never be smaller than the original disclosed quantity.

Any Iceberg order that is entered into the book below the iceberg minimum amount (as defined by MiFID II), or has its total amount updated to be below this amount, will be automatically converted to a Limit order. This conversion will be indicated to the clients in the **Ack** (03) message, by the *Ack Type* = Iceberg Transformed to Limit due to Minimum size.

2.3.9   Iceberg Order Partially Filled
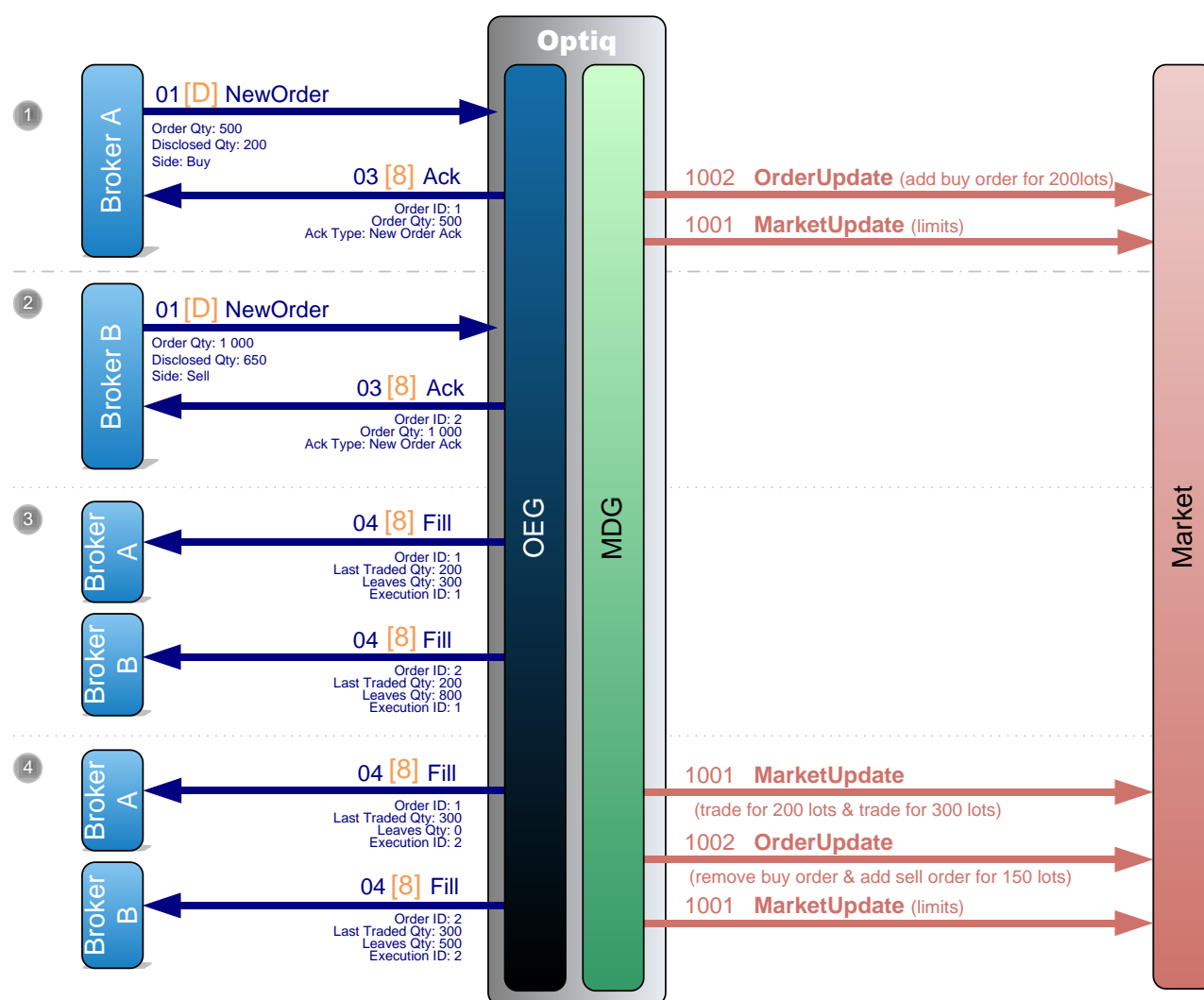
①   **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a total quantity of 500 and a disclosed quantity of 200.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

     1.3.0

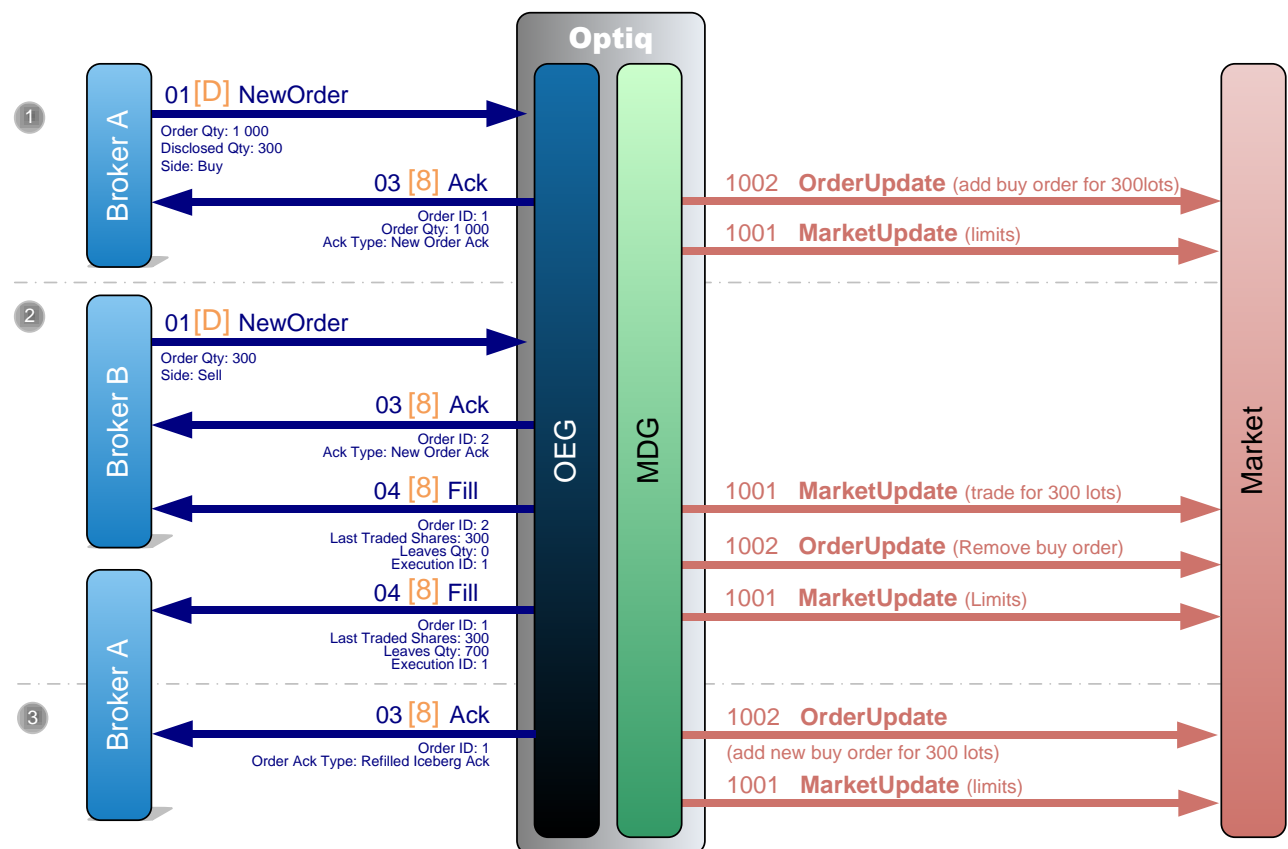The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order (qty=200) and a public **MarketUpdate** (1001) message to update the limit.

② **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a total quantity of 1,000 and a disclosed quantity of 650.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

③ The entering order immediately matches the first order for an initial quantity of 200 (Buy order disclosed quantity) and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution according to disclosed quantity.

④ The two orders then match for 300 (the remaining quantity of the Buy order) and OEG sends a private **Fill (04) (FIX 8)** message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for a trade of 200 lots, a trade of 300 lots along with a public **OrderUpdate** (1002) message to remove the first order (Buy) and add the Sell order for a disclosed quantity of 150 lots, followed by another **MarketUpdate** (1001) message to update the limits.

Note: Partially traded Iceberg orders are not subject to the checks of minimum quantity.

2.3.10 Breaching a Collar with Suspension

① **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with price=5, which is lower than the low static collar.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

② **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with price=5.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The instrument is automatically Reserved because of a potential trade outside of the collars. A public **MarketStatusChange** (1005) message is sent to the market.

A public **OrderUpdate** (02) (FIX D) message is sent to the market to add the order (Sell) in the order book and a public **MarketUpdate** (1001) message to update the limits.

③ Then a public **PriceUpdate** (1003) message is sent periodically to broadcast the Indicative Matching Price (IMP), as the instrument is suspended.

2.3.11 Breaching a Collar with Confirmation (No Reservation)

In the following example the instrument is configured with a Collar Logic applied to Blue Chips.

① **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 8,000 and a price=5, which is lower than the low collar.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

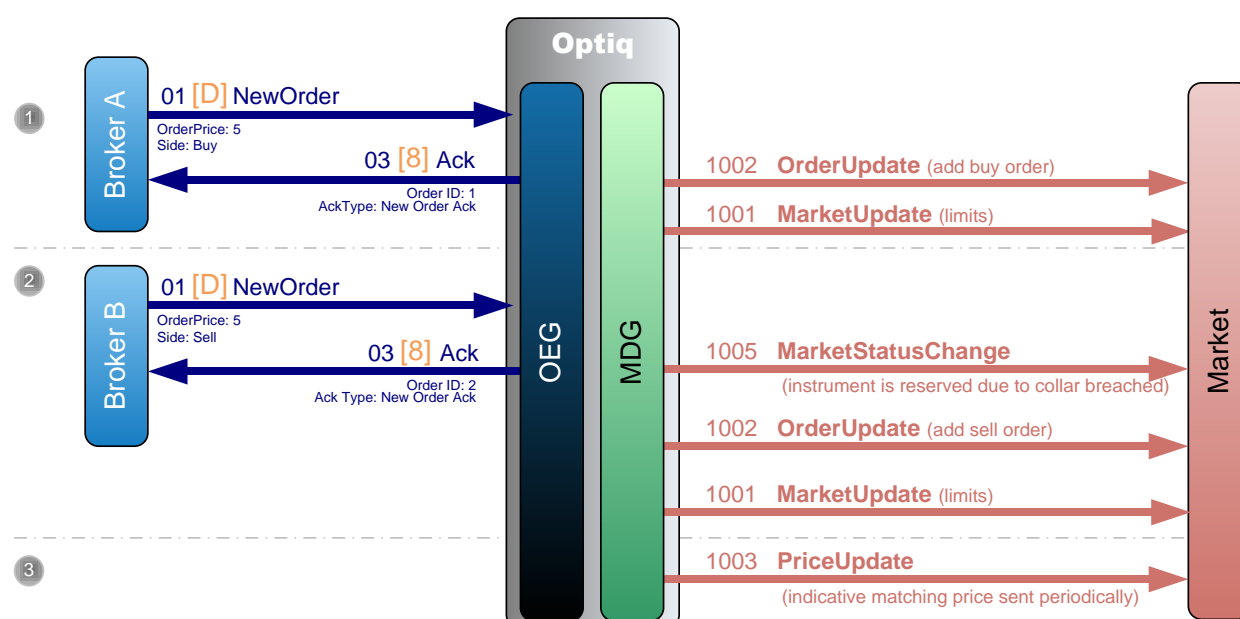The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

② **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 10,000 and a price=5.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order and private **Reject** (07) (FIX 9) message to reject/request confirmation for the order which is breaching the dynamic collar, and thus would match outside it.

③ **Broker B** sends a private **CollarBreachConfirmation** (20) (FIX G) message to confirm the order and OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical

processing of the order. (Please note that only one message should be sent to confirm, two messages are represented in the diagram because they are different for FIX and for SBE).

The dynamic collars are correspondingly updated around the low collar so the entering order matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for a trade and the new collars updated around this trade, along with a public **OrderUpdate** (1002) message to remove the first order (Buy) and add the sell order for its remaining quantity (2 000), followed by another **MarketUpdate** (1001) message to update the limits.

2.4   MODIFYING AN ORDER

2.4.1   Modifying an Unmatched Order
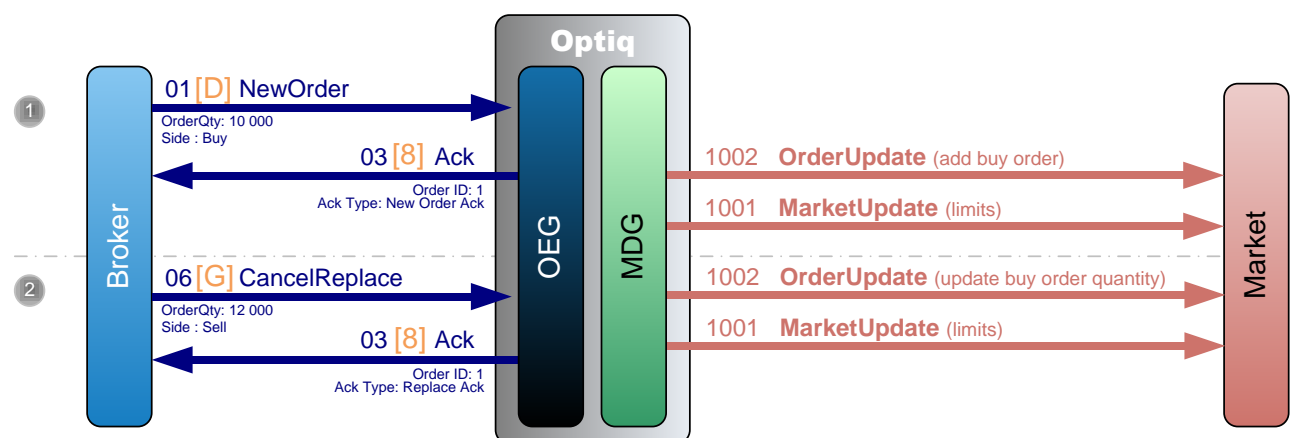
①   A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new buy order with a quantity of 10,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

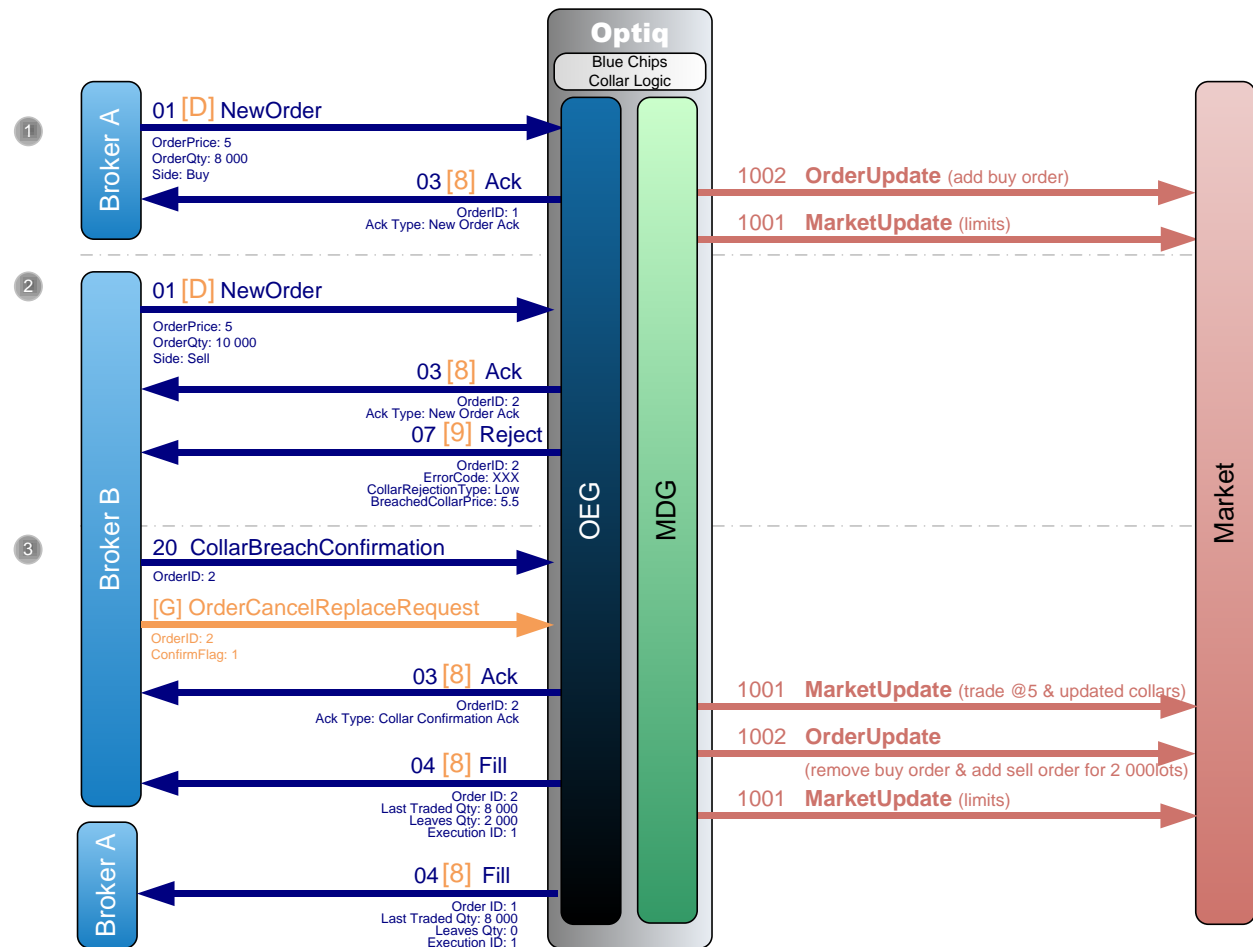The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

② The same Broker sends a private **CancelReplace** (06) (FIX G) message to modify the order by increasing the quantity up to 12,000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order modification.

MDG sends a public **OrderUpdate** (1002) message to the market to update the quantity of the previously entered order, followed by another **MarketUpdate** (1001) message to update the limits.

Note: In case of a change of an order ownership, i.e. when the **CancelReplace** (06) (FIX G) is sent from another OE Session, it will follows the same kinematic (no message is sent to the previous owner of the order).

2.4.2   Modifying a partially matched order

① **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 10 000.
OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

②   **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 8 000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

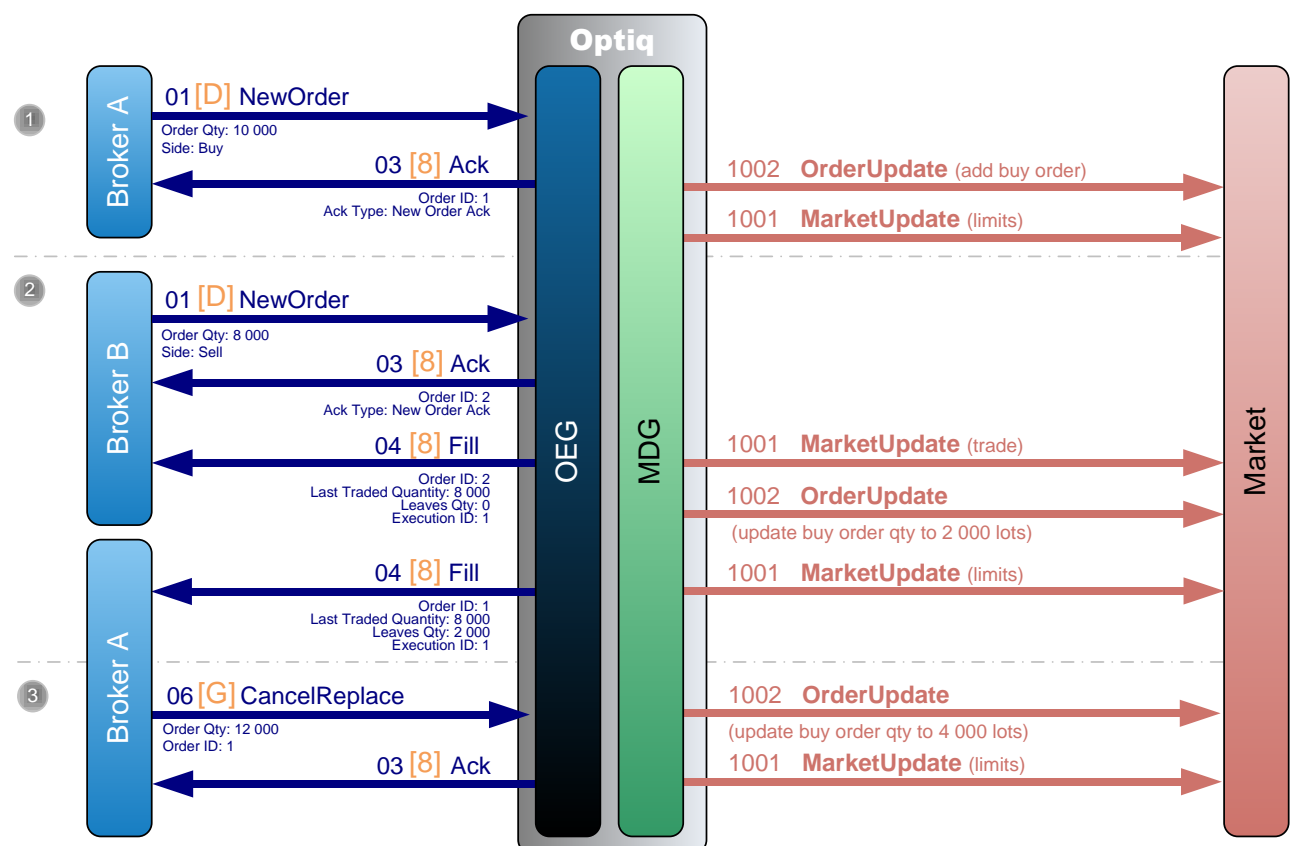The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for a trade along with a public **OrderUpdate** (1002) message to update the first order (Buy) quantity to 2 000, followed by another **MarketUpdate** (1001) message to update the limits.

③   Later, **Broker A** sends a private **CancelReplace** (06) (FIX G) message to modify the quantity of his Buy order. As he wants the leaves quantity to be equal to 4 000 after the modification, the broker indicates a quantity of 12,000 (as 8,000 have already matched).

OEG sends back a **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order modification.

A public **OrderUpdate** (1002) message is sent to the market to update the order quantity to 4 000 lots and a public **MarketUpdate** (1001) message to update the limit.

2.4.3   Rejected Modification

① **Broker A** sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 10 000.
OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

② **Broker B** sends a private **NewOrder** (01) (FIX D) message to enter a new Sell order with a quantity of 8 000.

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The entering order immediately matches the first order and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

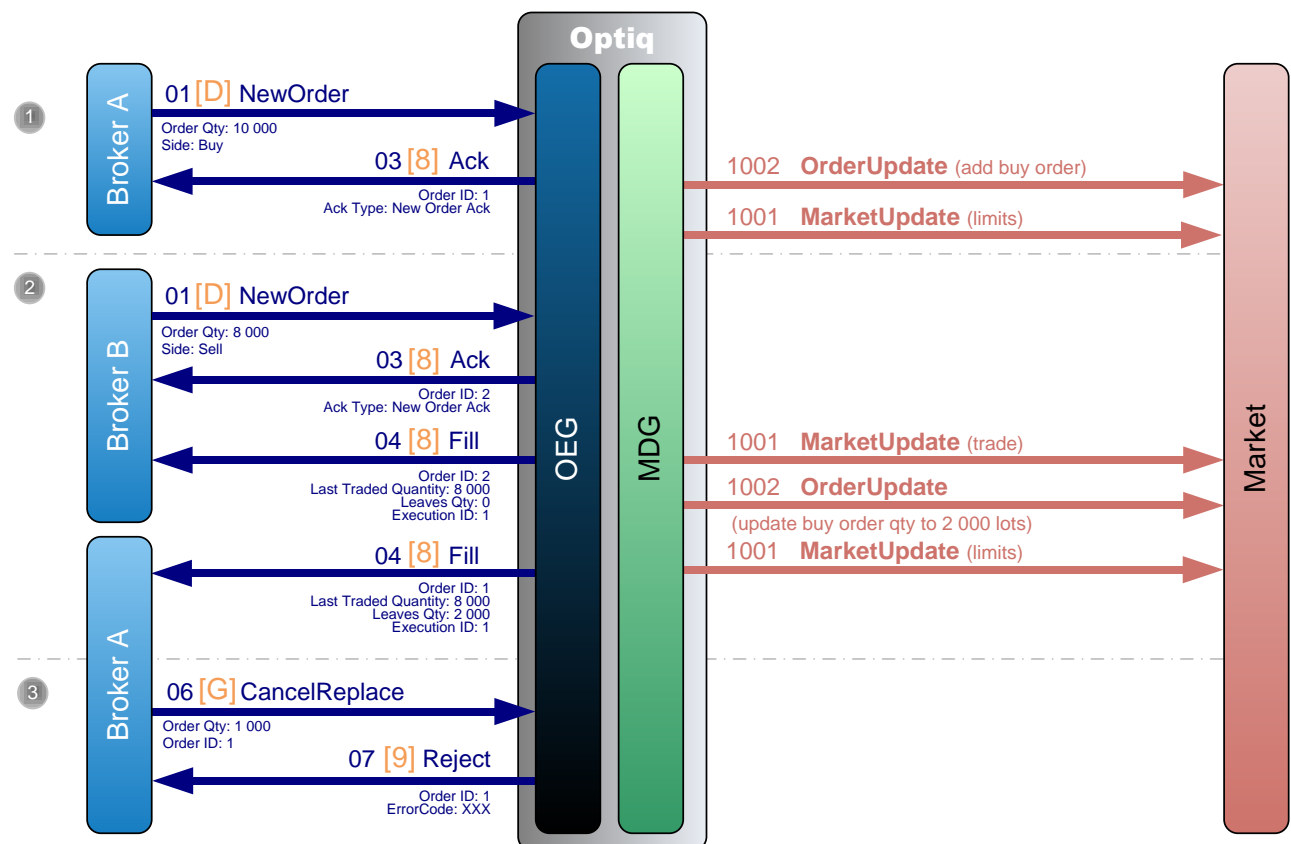A public **MarketUpdate** (1001) message is sent to the market for a trade along with a public **OrderUpdate** (1002) message to update the first order (Buy) quantity to 2 000, followed by another **MarketUpdate** (1001) message to update the limits.

③ Later, **Broker A** sends a private **CancelReplace** (06) (FIX G) message to modify the quantity of his Buy order. The broker indicates a quantity of 1 000.

OEG sends back a private **Reject** (07) (FIX 9) message to reject the replace operation as the quantity to be modified is no longer available. So the remaining quantity of 2 000 stays in the order book.

Note: If the broker attempts to change the quantity of an order to a value less or equal to the quantity already traded, the order modification will be rejected. In this example, new quantity of 8 000 will be rejected, a new quantity of 8 001 will be accepted.

2.5   CANCELLING AN ORDER
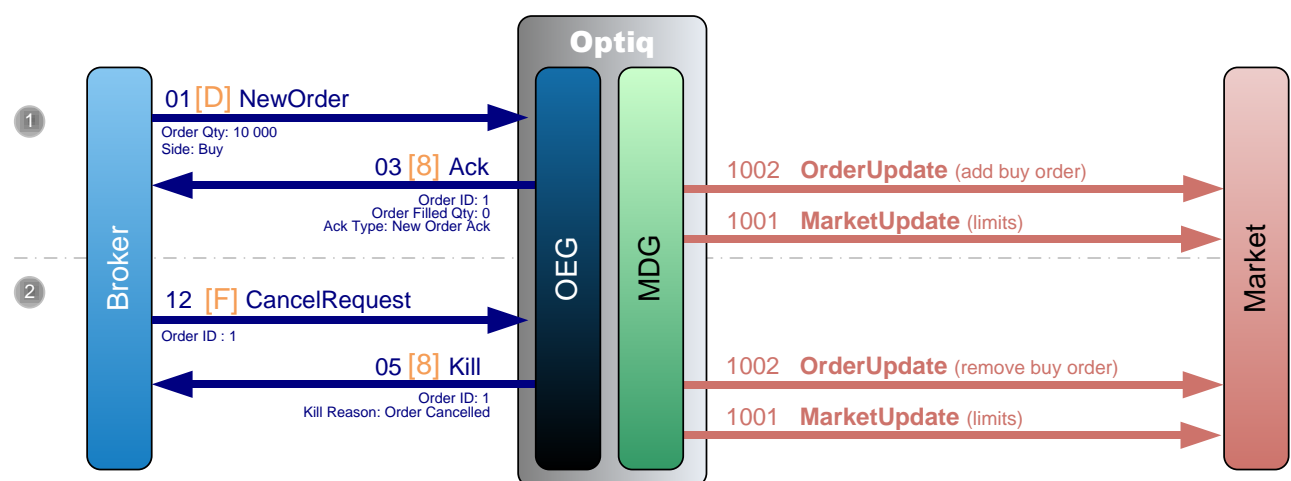
2.5.1   Cancelling an Unmatched Order

①   A Broker sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 10 000 lots.
OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

The order enters the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a public **MarketUpdate** (1001) message to update the limit.

②   Later the same Broker sends a private **CancelRequest** (12) (FIX F) message to cancel the previously entered order.

OEG sends back a private **Kill** (05) (FIX 8) message to confirm that the order request has been cancelled.

A public **OrderUpdate** (1002) message is sent to the market to remove the Buy, followed by another **MarketUpdate** (1001) message to update the limits.

2.5.2 Rejected Order Cancellation
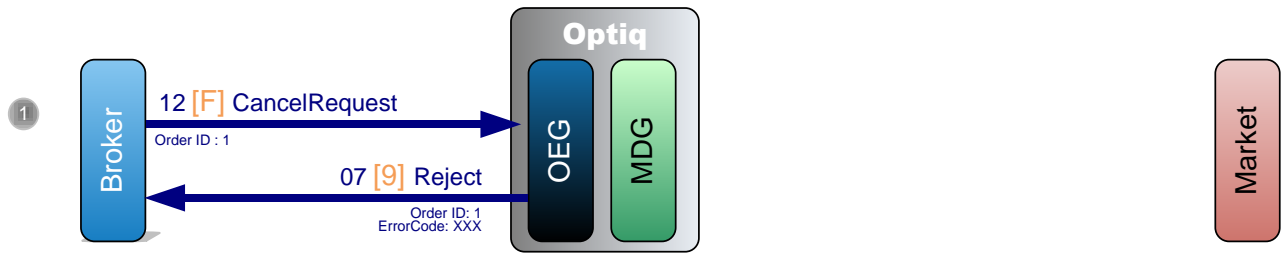
① A Broker sends a private **CancelRequest** (12) (FIX F) message to cancel an order that has already matched.

OEG sends back a private **Reject** (07) (FIX 9) message to reject the cancellation.

2.5.3 Mass Cancellation
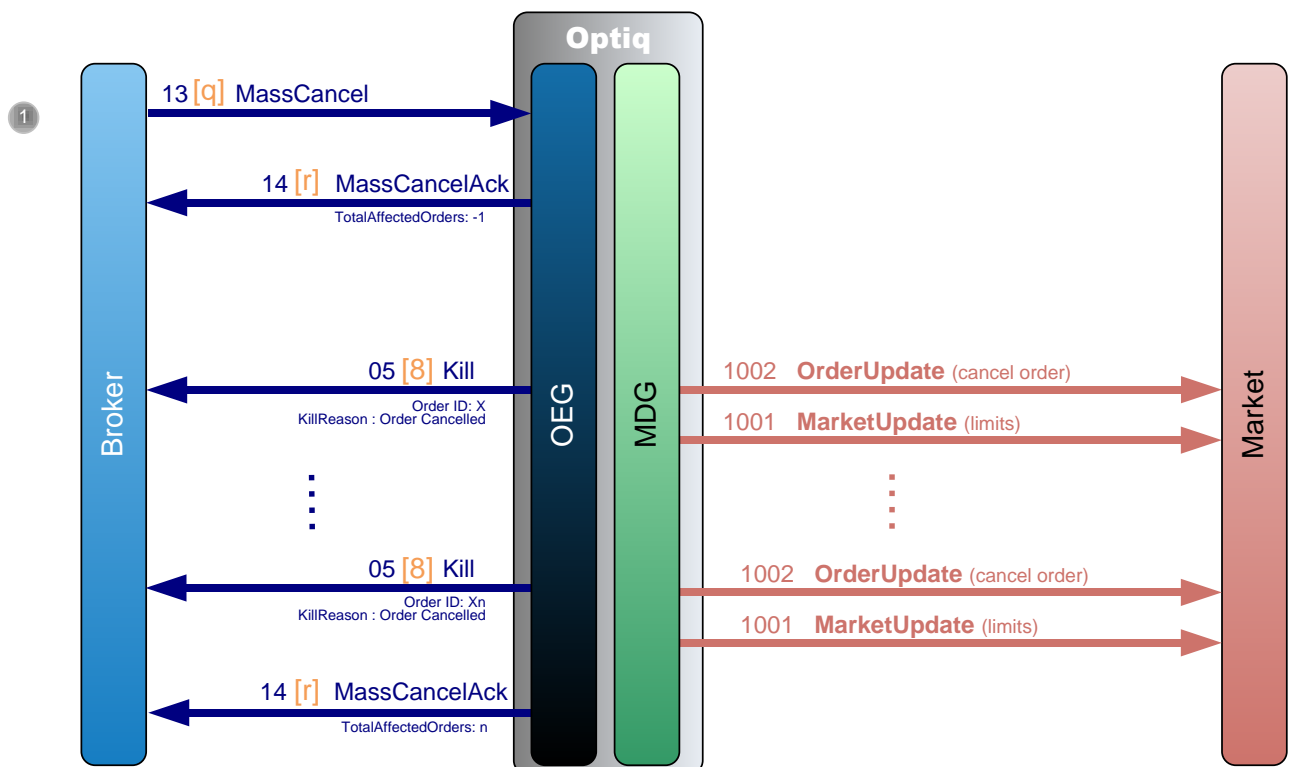
① A Broker sends a private **MassCancel** (13) (FIX q) message to cancel some of his orders matching specific criterions.

OEG sends back a private **MassCancelAck** (14) (FIX r) message followed by a private **Kill** (05) (FIX 8) message for each affected order detailing the killed orders; the mass cancellation process is ended by a new private **MassCancelAck** (14) (FIX r) message notifying the total affected orders.

A public **OrderUpdate** (1002) message is sent to the market to remove the killed orders along with a public **MarketUpdate** (1001) message to update the limits; both messages are sent for each affected instrument.

2.5.4   Cancel on Disconnect Mechanism
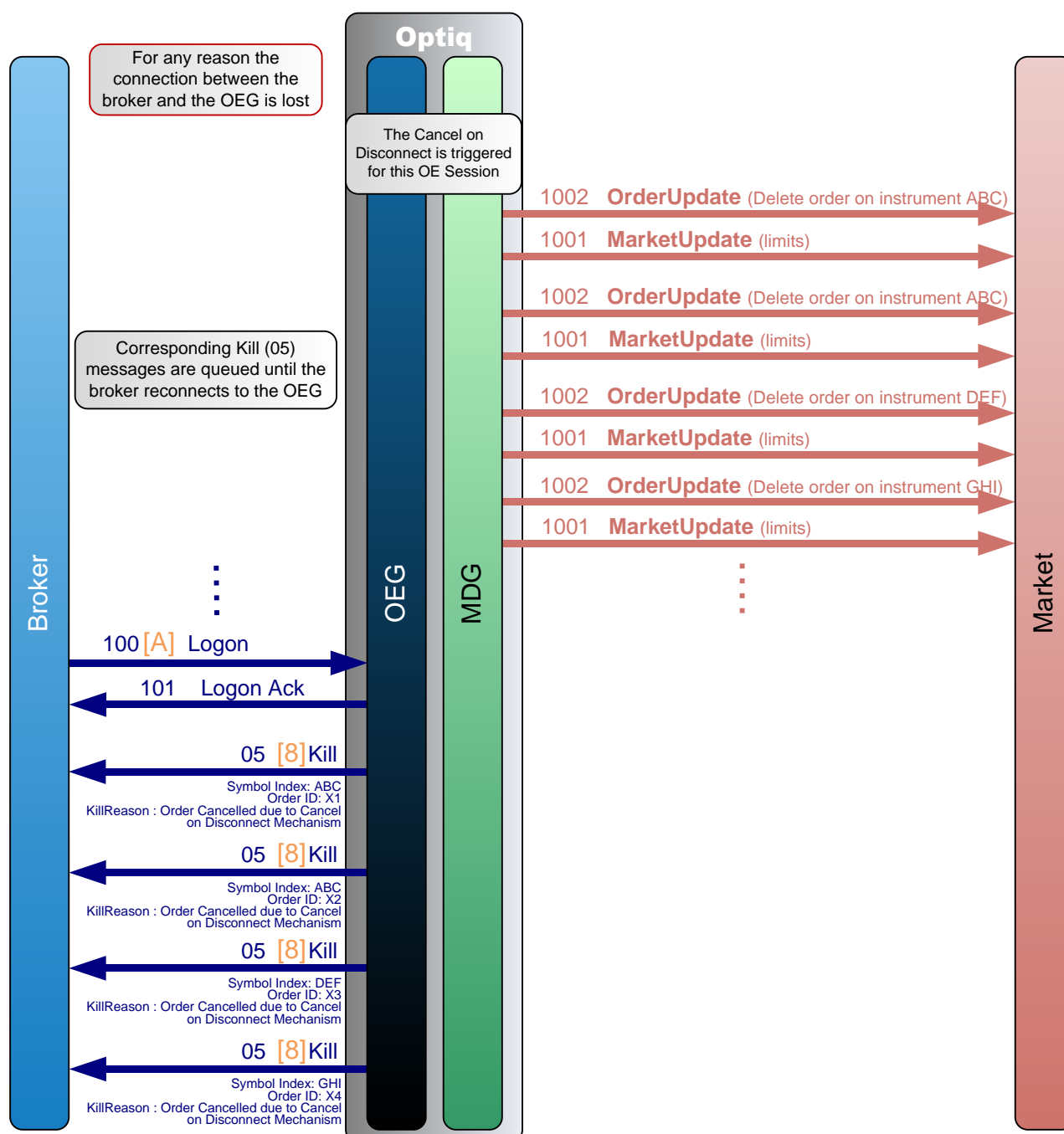
The diagram represents a generic case of loss of connection (physical) between a client and a partition.

When a connection is lost between the broker and OEG, for any reason, the Cancel on Disconnect (CoD) mechanism is triggered for all OE Sessions concerned by the connection outage. Once the mechanism is triggered, all live orders not flagged to be persisted and belonging to the

corresponding OE Session(s) are immediately cancelled for their remaining quantity, regardless of order type and validity type.

For each order cancelled a public **OrderUpdate** (1002) message is sent to the market to remove the order and a public **MarketUpdate** (1001) message to update the limits.

For each cancelled order a **Kill** (05) (FIX 8) message is generated and queued until the client reconnects.

When the Broker reconnects with a **Logon** (100) (FIX A) message, if the logon is successful the OEG sends back a **LogonAck** (101) (FIX A) message.

Once the connection is re-established, the Broker immediately receives the **Kill** (05) (FIX 8) messages that have been queued.

Note: Scope of Cancel on Disconnect only includes orders sent during the current day. Orders entered during a previous business day are not in scope of Cancel on Disconnect and are not impacted.

2.6   OWNERSHIP REQUEST

2.6.1   Ownership request for a specified order ID
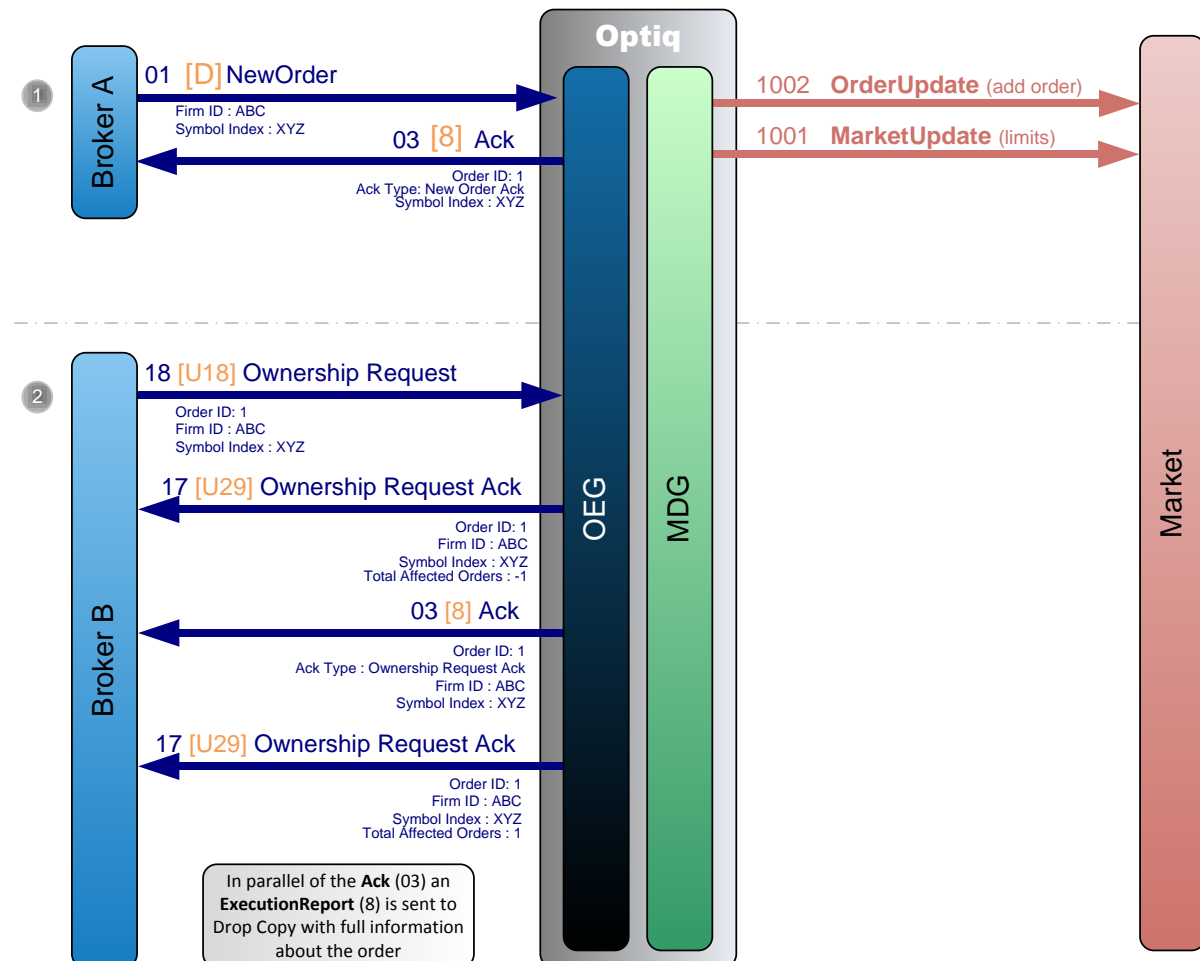
①   A Broker A sends a private **NewOrder** (01) (FIX D) message to enter a new order.

OEG sends back an **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order (*Order ID* = 1).

The order enters into the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

②   Another participant, Broker B sends an **Ownership Request** (18) (FIX U18) to request the ownership of the previous order (*Order ID* = 1) sent by Broker A.

OEG sends back an **Ownership Request Ack** (17) (FIX U29) message to Broker B, to confirm the reception of the request (with *Total Affected Orders* = -1).

OEG sends back an **Ack** (03) (FIX 8) message to Broker B, to give the detail of the order (*Order ID* = 1). In parallel of the **Ack** (03) (FIX 8) an **ExecutionReport** (8) is sent to Drop Copy with full information about the order.

OEG sends back another **Ownership Request Ack** (17) (FIX U29) message to Broker B, to confirm the successful change of ownership of the order (*Order ID* = 1) from Broker A's OE session to Broker B's OE session (*Total Affected Orders* = 1). Broker A does not receive any messages of this exchange and following the transfer of ownership all unsolicited messages for the affected order are sent to Broker B.

2.6.2   Ownership request for a Logical Access or OE Session
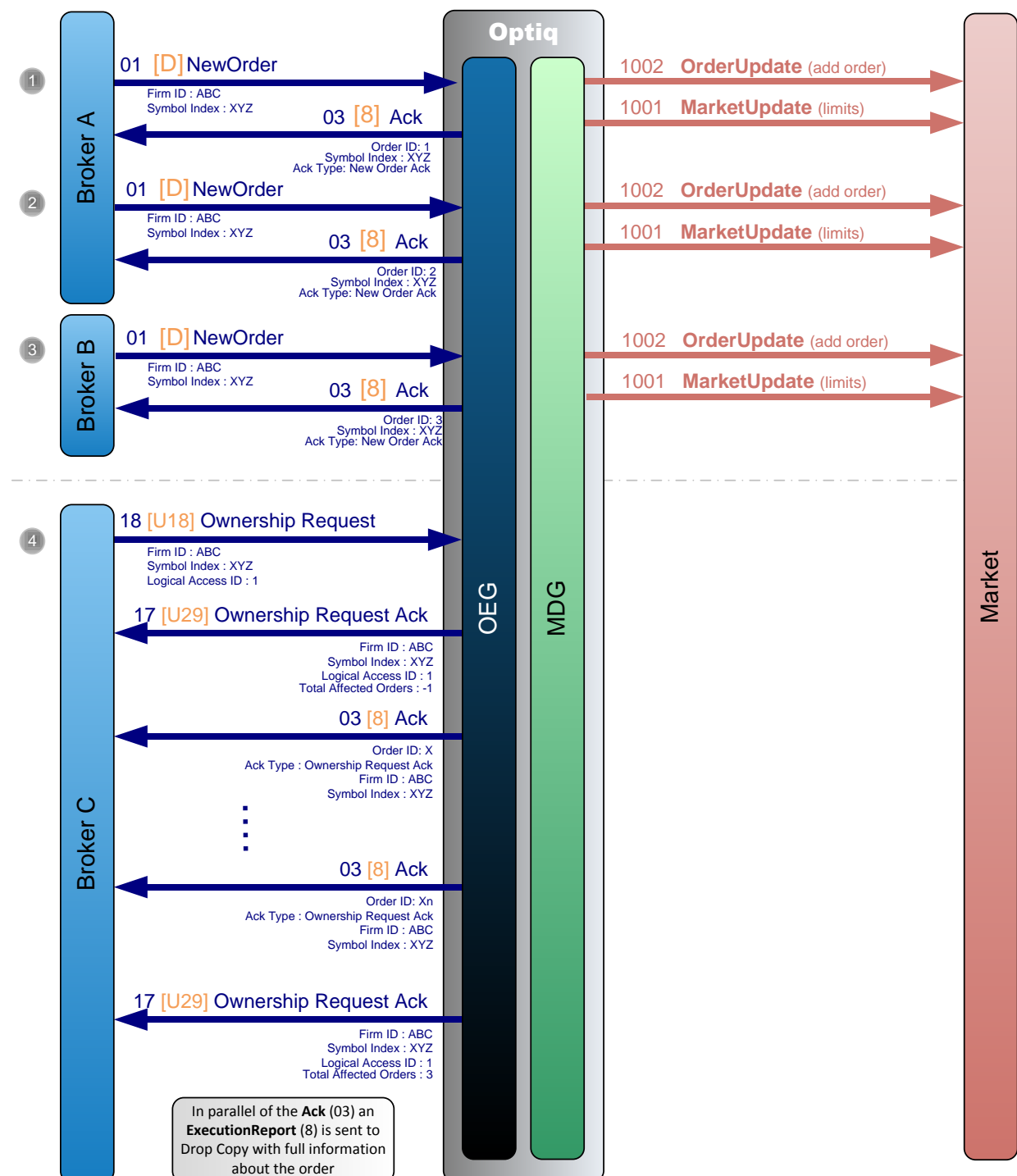
Logical Access ID and *OE Session ID* are provided by clients in the **Logon** (100) message.

① A Broker A sends a private **NewOrder** (01) (FIX D) message to enter a new order on instrument XYZ.

The order is entered through the OE session 1A (Logical Access ID = 1, OE Partition ID = A).

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order (*Order ID* = 1).

The order enters into the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

② Broker A sends another private **NewOrder** (01) (FIX D) message to enter a new order instrument XYZ.

The order is entered through the OE session 1A (Logical Access ID = 1, OE Partition ID = A).

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order (*Order ID* = 2).

The order enters into the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

③ Broker B sends a private **NewOrder** (01) (FIX D) message to enter a new order instrument XYZ.

The order is entered through the OE session 1B (Logical Access ID = 1, OE Partition ID = B).

OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order (*Order ID* = 3).

The order enters into the order book without matching and a public **OrderUpdate** (1002) message is sent to the market to add the order and a **MarketUpdate** (1001) message to update the limits.

④ Another participant, Broker C sends an **Ownership Request** (18) (FIX U18) to request the ownership of the orders of the *Logical Access ID* = 1 for the instrument XYZ.

The request is entered through the OE session 2A (Logical Access ID = 2, OE Partition ID = A).

OEG sends back an **Ownership Request Ack** (17) (FIX U29) message to Broker C, to confirm the reception of the request (with *Total Affected Orders* = -1).

OEG sends back an **Ack** (03) (FIX 8) message to Broker C for each order (*Order ID* = 1, 2 and 3) for the instrument XYZ that are owned by the Logical Access 1. In parallel of each **Ack** (03) (FIX 8) message an **ExecutionReport** (8) is sent to Drop Copy with full information about the order.

OEG sends back another **Ownership Request Ack** (17) (FIX U29) message to Broker C to confirm the successful change of ownership of the orders belonging to the Logical Access ID =1 for the instrument XYZ. The ownership of *Order ID* = 1 and 2 from Broker A's OE session and *Order ID* = 3 from Broker B's OE session transfer to Broker C's OE session (*Total Affected Orders* = 3). Brokers A and B do not receive any messages of this exchange, and following the transfer of ownership all unsolicited messages for the affected orders are sent to Broker C.

Note: All specified Logical Access IDs and OE Sessions belong to the same Firm.

2.7 OPENING/UNCROSSING

2.7.1 Market To Limit on Opening
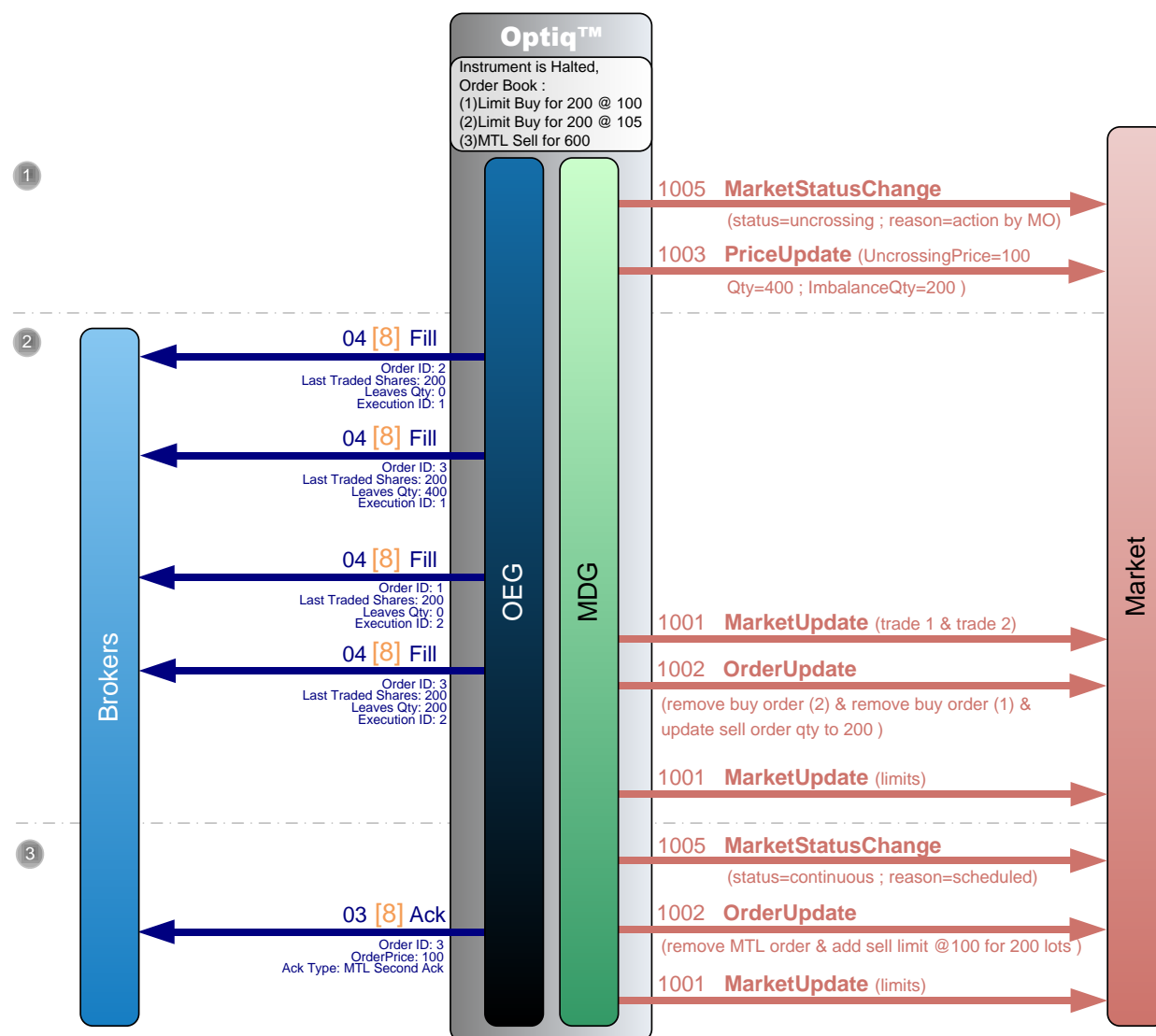
There are already two Limit orders in the order book, one Buy at price=100 and quantity=200 and another Buy at price=105 and quantity=200. And there is also one Sell Market To Limit order with quantity=600. The last traded price is 100.

① The instrument reopens (is unsuspended). A first public message **MarketStatusChange** (1005) message is sent to the market to notify the resumption of trading on the instrument along with a public **PriceUpdate** (1003) message to broadcast the Uncrossing Price.

② The Sell order matches with the best Buy order (2) for 200 and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution. Then the Sell order matches with the Buy order (1) for 200 and OEG sends back a private **Fill** (04) (FIX 8) message to each broker to notify the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for both trades along with a public **OrderUpdate** (1002) message to remove the Buy order (2) and the Buy order (1) and update the quantity of the Sell order to 200 lots.

At the end of the uncrossing processing MDG sends a **MarketUpdate** (1001) message to broadcast the updated values of each limit that has changed during the uncrossing, or only the BBO depending on the market.

③   A public **MarketStatusChange** (1005) message is sent to the market to indicate that the instrument is now in a continuous phase.

Directly after the change of phase, a public **OrderUpdate** (1002) message is sent to the market to modify the Sell MTL order to a Sell Limit order at 100 for 200 lots. (when the instrument switches to a Continuous trading phase, all MTL orders partially or not executed always become Limit orders at the uncrossing price for their remaining quantity).

An **Ack** (03) message is sent to the broker with *Ack Type* = MTL Second Ack, for the transformation of the MTL order into a Limit order.

A public **MarketUpdate** (1001) message is sent to the market to update the limits.

Note: At the beginning of each uncrossing processing a **PriceUpdate** (1003) message is sent to the market to broadcast the uncrossing price (even if it is equal to the last IMP broadcasted). During each uncrossing processing there is always only one public **MarketUpdate** (1001) message sent to the market to update the limits, which is sent at the very end of the processing.

2.8   Indicative Price Inputs

Indicative price inputs include:

- Valuation Price.
- Alternative Indicative Price.

2.8.1   Valuation Price by Liquidity Provider Price Input message
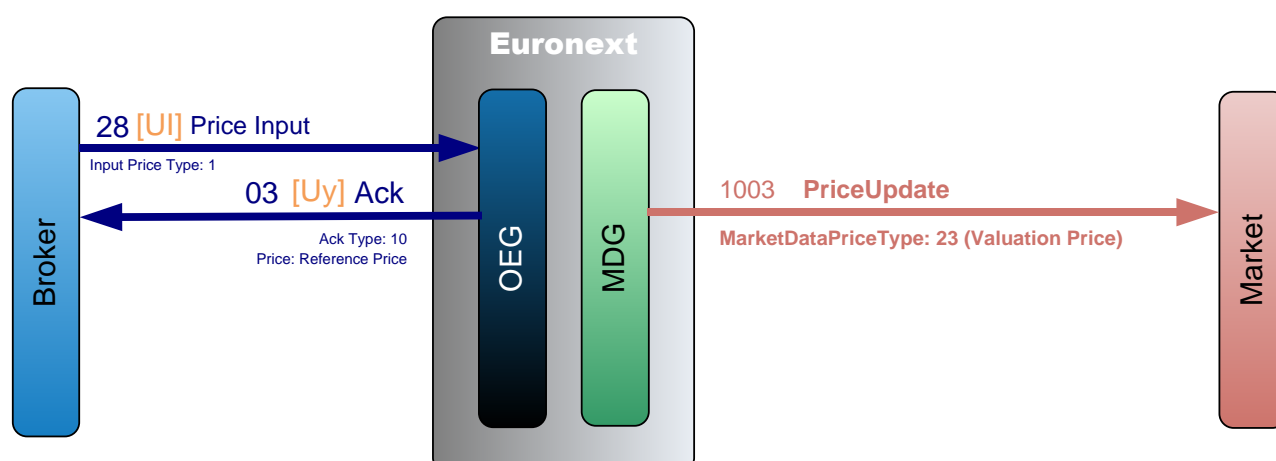
A price information message can be disseminated to Market Participants in order to valuate some illiquid bonds instruments through an Alternative Indicative Price (AIP).

The Liquidity Provider sends a private **Price Input** (28) (FIX UI) message with *Input Price Type = '1' (Valuation Trade)* without specifying any price.

The valuation price takes the value of the current reference price of the warrant instrument and a public **PriceUpdate** (1003) message is sent to the Market participants to disseminate it.

Note: No price is specified in the Valuation Price Input message.

2.8.2 AIP (Alternative Indicative Price) using Member Firm Price
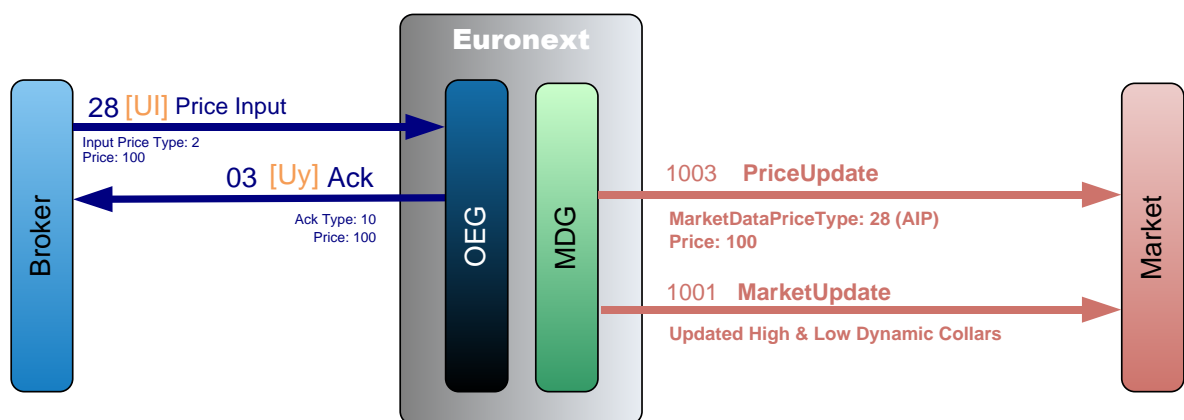
A price information message can be disseminated to Market Participants in order to valuate some illiquid bonds instruments through an Alternative Indicative Price (AIP).

The Broker sends a specified price using a private **Price Input** (28) (FIX UI) message with *Input Price Type = '2' (Alternative Indicative Price)*.

This price is then generated and disseminated to the Market participants through a public **PriceUpdate** (1003) message.

The Dynamic Collars of the Instrument are updated around the price specified in the **Price Input** (28) (FIX UI) message and a public **MarketUpdate** (1001) message is sent to the market to disseminate the new collars.

3.   UNSOLLICITED MESSAGES

3.1   ASYNCHRONOUS MESSAGES

3.1.1   Statistics Message

The public **Statistics** (1009) message is sent to the market after each trade, it includes only the information that needs to be updated. It can include minimum and maximum traded prices for daily, yearly and lifetime periods along with the cumulative volume since the start of the trading day and the percentage of variation of the traded price versus the last reference price.

3.1.2   Automatic IMP Calculation
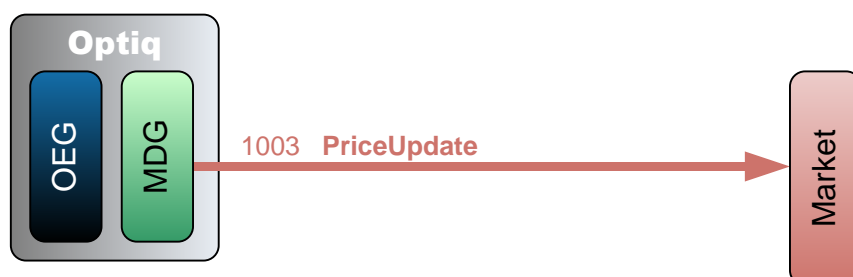
A public **PriceUpdate** (1003) message is sent periodically to the market when the instrument is in a Call or Suspended phase to broadcast the Indicative Matching Price (IMP).

Note: The IMP is broadcasted only if the IMP price or quantity have changed from the values previously sent.

3.2   ACTIONS PERFORMED BY MARKET OPERATIONS

3.2.1   Reference Price Update
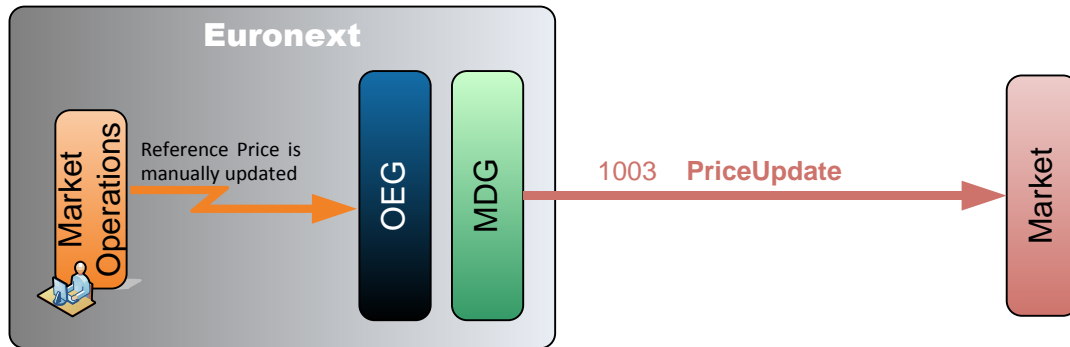
Market Operations send a private command to Optiq to update the reference price on the given instrument.

A manual update by Market Operations of the reference price on a given instrument covers:

- ◆ Dynamic Collar Reference Price modification by Market Operations,
- ◆ Static Collar Reference Price modification by Market Operations,
- ◆ Last Adjusted Closing Price modification by Market Operations,
- ◆ Last Traded Price modification by Market Operations command (if the Dynamic Collar Reference Source is not 'External').

Optiq sends a public **PriceUpdate** (1003) message to broadcast the new prices.

3.2.2 Bulk Order Cancellation
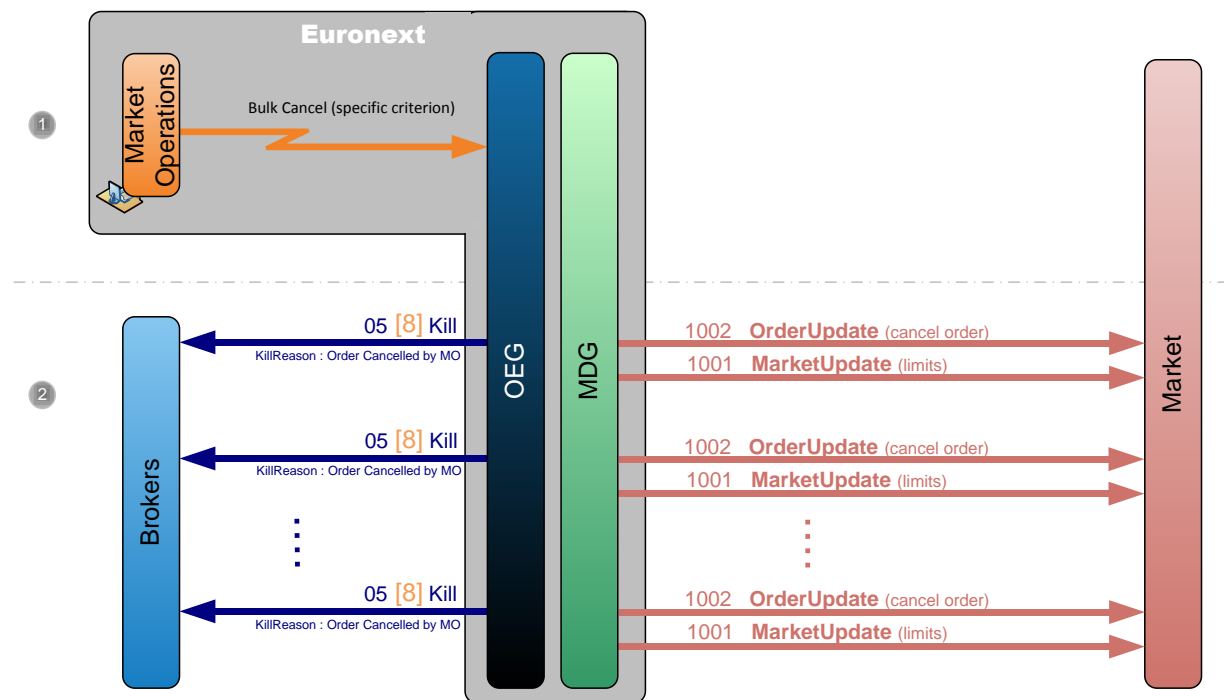
①   Market Operations cancel orders matching a specified criterion.

②   Optiq sends a private **Kill** (05) (FIX 8) message for each cancelled order to the broker who entered the order, and as many public **OrderUpdate** (1002) and **MarketUpdate** (1001) messages per instrument to the market to respectively remove the cancelled orders and update the limits.

3.2.3 Trade Cancellation
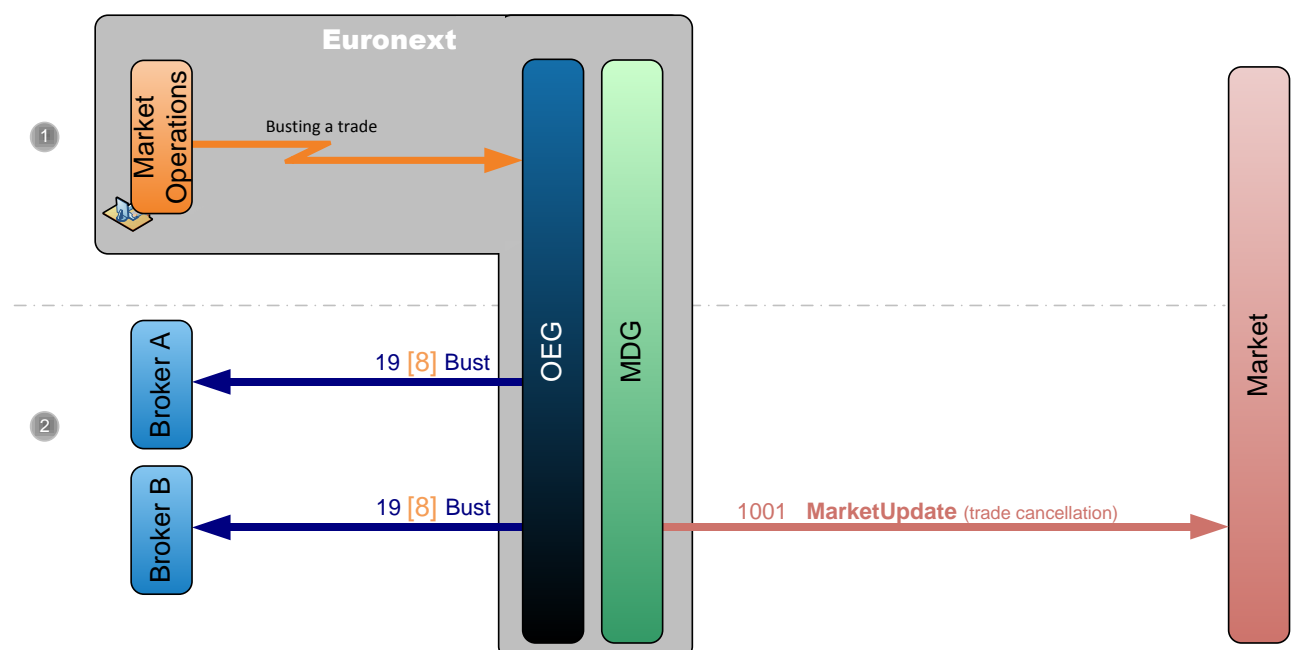
① Market Operations busts a trade on behalf of two brokers.

② Optiq sends a private **TradeBustNotification** (19) (FIX 8) message for the cancelled trade to the brokers who entered the orders and a public **MarketUpdate** (1001) message to remove the cancelled orders.

3.2.4   Suspending an Instrument
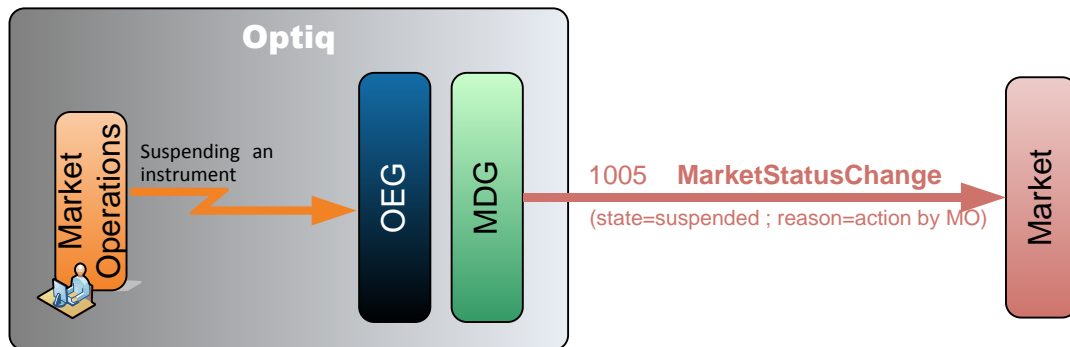
Market Operations suspends a specific instrument.

Optiq sends a public **MarketStatusChange** (1005) message to the market to indicate that the instrument has been suspended.

4.　　MARKET STATUS CHANGES

This section is dedicated to all market publications that deal with changes of Market Status on Euronext Cash markets, which are communicated via the **MarketStatusChange** (1005) message.

The Optiq MDG **MarketStatusChange** (1005) is common across all Euronext Markets, including Cash, Warrants and off-book data publication. For all markets, the Market Status of an instrument can be determined using the following fields:

- **Instrument State**: Market State of the Instrument
- **Status Reason**: Instrument State origin
- **Trading Mode**: Specifics during a trading phase that do not impact the Instrument State
- **Trading Period**: indicates the trading period
- **Trading Side**: indicates the side of a One-Side Only period for LP Quote Driven Warrant Market Model
- **Order Entry Qualifier**: Describes whether order entry is allowed for the instrument, and depends on Instrument State, Trading Mode and global availability
- **Scheduled Event**: Market Event notification
- **Scheduled Time**: Scheduled Event associated time if required

The possible Market Status values on Euronext Cash are as follows:

MDG MarketStatusChange Repeating Section

MarketDataChangeType
0 Status Change(s)
1 Scheduled Event Notification
2 Status Change(s) & Scheduled Event Notification

SymbolIndex

EventTime

BookState
1 Inaccessible
2 Closed
3 Call
4 Uncrossing
5 Continuous
6　Halted
7 Continuous Uncrossing
8 Suspended
9 Reserved

StatusReason
0 Scheduled
4 Collars breached
7 Automatic Reopening
8 No Liquidity Provider
11 Knock-In By Issuer
12 Knock-Out By Exchange
13 Knock-Out By Issuer
15 Action by MO
20 New Listing
21 Due to Underlying
22 Outside of LP quotes
23 Technical

PhaseQualifier
0 No Qualifier
1 Call BBO Only
2 Trading At Last
3 Random Uncrossing

TradingPeriod
1 Opening
2 Core
3 Closing

TradingSide
1 Bid Only
2 Offer Only
3 PAKO
4 Both Sides

OrderEntryQualifier
0 OE/Cancel/Modify Disabled
1 OE/Cancel/Modify Enabled
3 Cancel Only

ScheduledEvent
0 Cancel Previously Scheduled Event
1 Reopening
3 Resumption of trading

ScheduledEventTime

One of the main improvements of this message is that it always provides the full state of the instrument. The updated instrument state values will appear accordingly in the updated fields, and the unchanged values from the previous state will persist. Using this method, no interpretation as all required information is provided each time the message is sent.

In the following Market Status change example, an instrument is manually suspended by Market Operations with Order entry disabled:

1005　**MarketStatusChange**

BookState: 8 (Suspended)
StatusReason: 15 (Action by MO)
TradingPeriod: Current Value
OrderEntryQualifier: 0 (OE/Cancel/Modification disabled)
ScheduledEvent: N/A

In the following diagrams some values of the **MarketStatusChange** (1005) message are set to 'Current Value'. It means that the value is the same as the one sent in the previous **MarketStatusChange** (1005) message. In fact the value will be populated in the message, here they are set to 'Current Value' to highlight that previous values that are still valid are populated even if they have not changed.

4.1 AUTOMATIC MARKET STATUS CHANGES

4.1.1 Scheduled Uncrossing
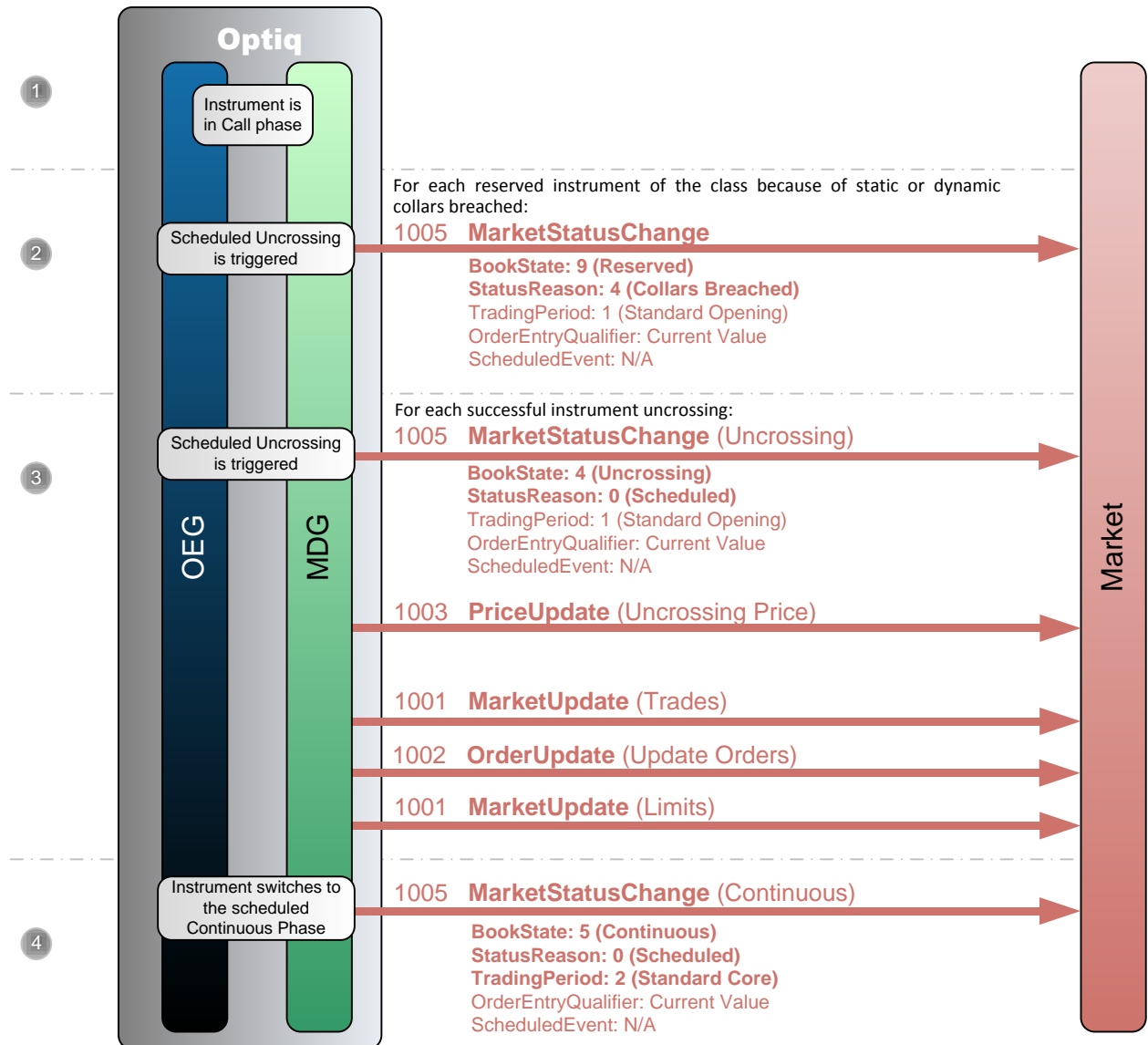
① The instrument is in a Call trading phase as defined in the **TimeTable** and by the pattern associated to this instrument. (Please note that all the scheduled state changes are notified and described in the pattern of the instrument.)

② When the Uncrossing is triggered if the uncrossing price is outside of the collars (the uncrossing price is greater than the High Collar for example) the instrument is automatically Reserved and a **MarketStatusChange** (1005) message is disseminated to the market.

③ When the Uncrossing is triggered if the uncrossing price lies within the collars the uncrossing is performed and a public **MarketStatusChange** (1005) message is disseminated to the market.

Note: In the case of a Blue Chip uncrossing, the *Trading Mode* value is '2' (Random Uncrossing).

Right after the status change a public **PriceUpdate** (1003) message is sent to the market with the uncrossing price and the quantity at which the uncrossing is performed.

For each trade generated a public **MarketUpdate** (1001) is sent for the trade along with a public **OrderUpdate** (1002) message to update the corresponding orders.

At the end of the uncrossing process a public **MarketUpdate** (1001) message is sent to update the values of each limit that has changed.

④   A public **MarketStatusChange** (1005) message is sent to the market to indicate that the instrument is now in a continuous phase.

4.1.2   Scheduled Continuous TAL

① Instrument is in Continuous phase

② Instrument switches to the scheduled Call phase

1005 **MarketStatusChange**

BookState: 3 (Call)
StatusReason: Current Value 0 (Scheduled)
TradingPeriod: 3 (Closing)
OrderEntryQualifier: Current Value 1 (OE/Cancel/Modify enabled)
ScheduledEvent: N/A

Valid For Closing (VFC) orders are triggered as the trading period is now *Closing*:

1002 **OrderUpdate** (add order)

1001 **MarketUpdate** (limits)

⋮

1002 **OrderUpdate** (add order)

1001 **MarketUpdate** (limits)

The Indicative Matching Price is dessiminated periodically:

1003 **PriceUpdate** (IMP)

Other orders can enter the book:

1002 **OrderUpdate** (add order)

1001 **MarketUpdate** (limits)

③ Scheduled Uncrossing is triggered

1005 **MarketStatusChange** (uncrossing)

BookState: 4 (Uncrossing)
StatusReason: Current Value 0 (Scheduled)
TradingPeriod: Current Value 3 (Closing)
OrderEntryQualifier: Current Value
ScheduledEvent: N/A

1003 **PriceUpdate** (Uncrossing Price)

1001 **MarketUpdate** (Trades)

1002 **OrderUpdate** (Update Orders)

1001 **MarketUpdate** (Limits)

④ Instrument switches to the scheduled Continuous TAL phase

1005 **MarketStatusChange**

BookState: 5 (Continuous)
StatusReason: Current Value 0 (Scheduled)
TradingPeriod: Current Value 3 (Closing)
OrderEntryQualifier: Current Value
ScheduledEvent: N/A
TradingMode: **Trading At Last**

OEG   MDG   Market

①   The instrument is in a Continuous trading phase as defined in the **TimeTable** and by the pattern associated to this instrument. (Please note that all the scheduled state changes are notified and described in the pattern of the instrument.)

②   When the instrument switches to the Call trading phase a public **MarketStatusChange** (1005) message is disseminated to the market.

      As the trading period is now *Standard **Closing*** the VFC and VFU orders are triggered and enter the order book. For each order entering the book a public **OrderUpdate** (1002) to add the order is sent to the market along with a public **MarketUpdate** (1001) message for the limits.

      During the Call phase the IMP is disseminated periodically with a public **PriceUpdate** (1003) message.

      During the Call phase orders can be entered, modified and cancelled. It will affect the value of the IMP.

③   When the Uncrossing is triggered if the uncrossing price lies within the collars the uncrossing is performed and a public **MarketStatusChange** (1005) message is disseminated to the market.

      Right after the status change a public **PriceUpdate** (1003) message is sent to the market with the uncrossing price and the quantity at which the uncrossing is performed.

      For each trade generated a public **MarketUpdate** (1001) is sent for the trade along with a public **OrderUpdate** (1002) message to update the corresponding orders.

      At the end of the uncrossing process a public **MarketUpdate** (1001) message is sent to update the values of each limit that has changed.

④   A public **MarketStatusChange** (1005) message is sent to the market to indicate that the instrument is now in a continuous Trading At Last (TAL) phase.

4.2   Market Status Changes Due To Manual Intervention

4.2.1   Instrument Suspended by Market Operations
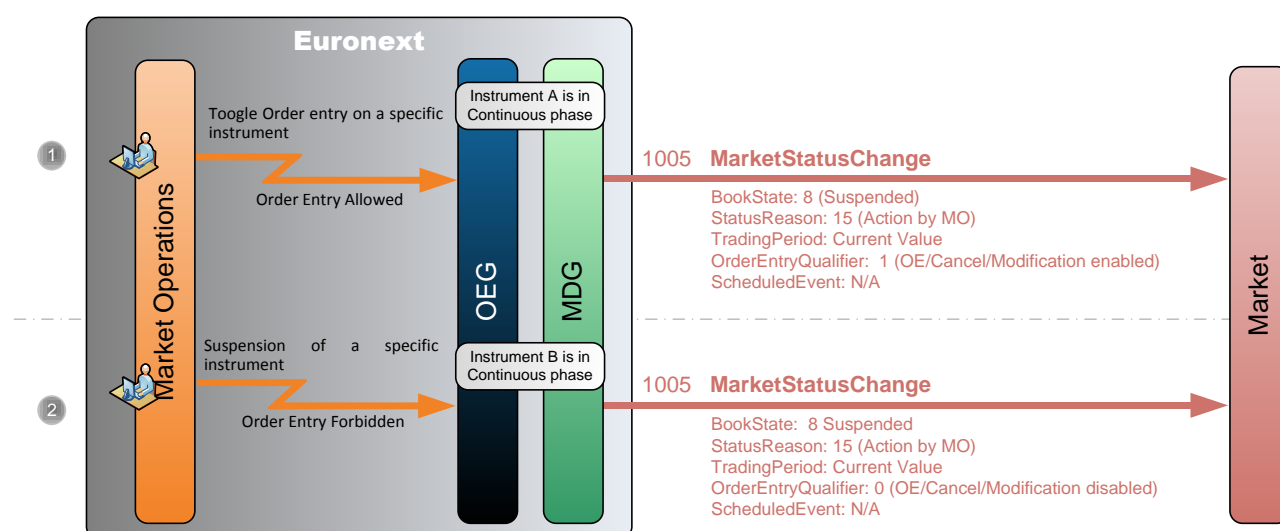
① The instrument A is in a Continuous trading phase as defined in the **TimeTable** and by the pattern associated to this instrument.
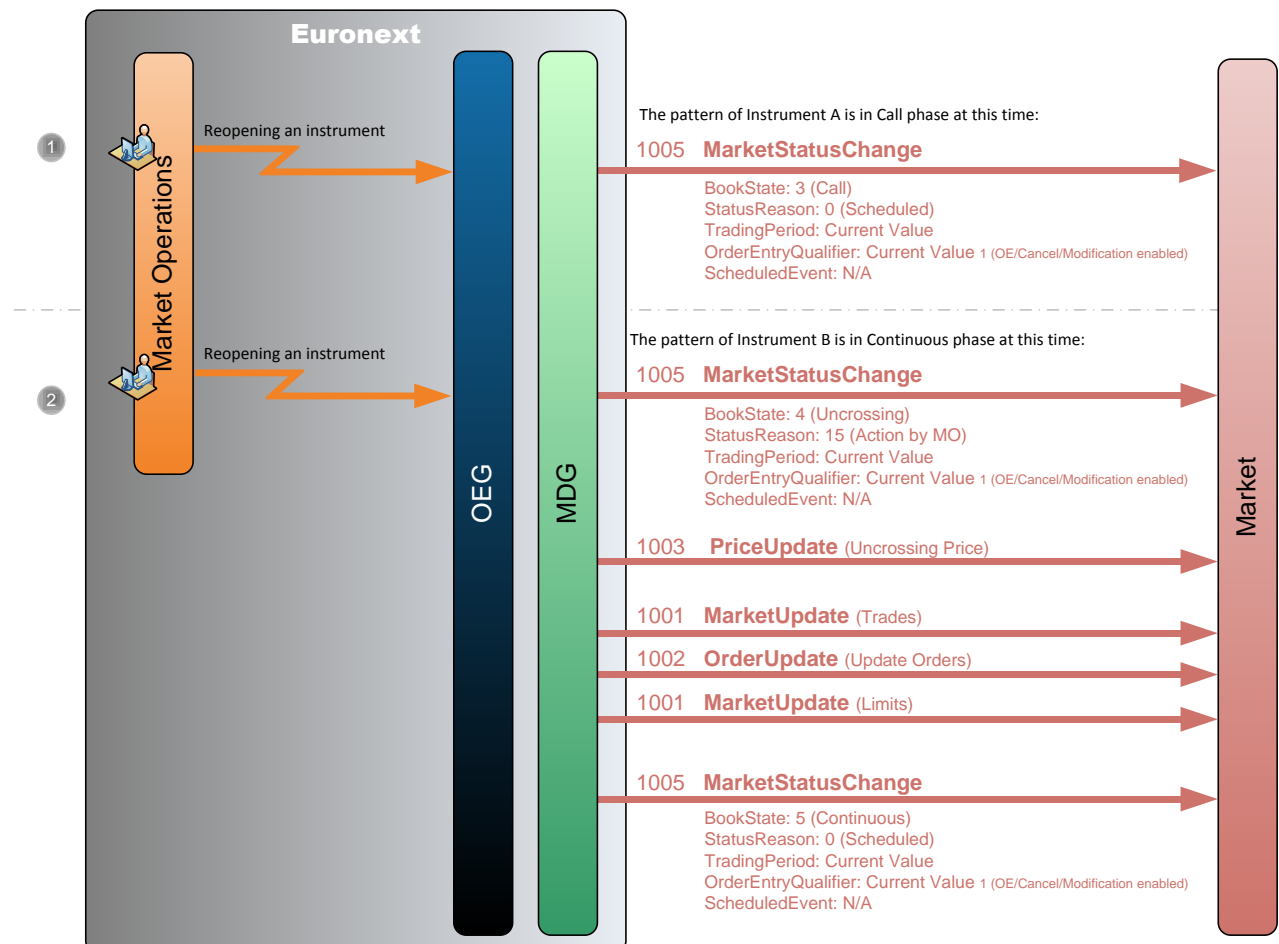
Market Operations suspends the instrument and let the Order Entry enabled, it is notified to the market by a public **MarketStatusChange** (1005) message.

② The instrument B is in a Continuous trading phase as defined in the **TimeTable** and by the pattern associated to this instrument.

Market Operations suspends the instrument and disables the Order Entry, it is notified to the market by a public **MarketStatusChange** (1005) message.

Note: The IMP is no longer disseminated if the phase was a Call phase, and no trading is possible in Continuous phase.

4.2.2 Instrument Reopened by Market Operations

① Market Operations reopens the instrument A, hence the instrument comes back to a Call trading phase as defined by its pattern at this time. It is notified to the market by a public **MarketStatusChange** (1005) message.

②   Market Operations reopens the instrument B, hence the instrument will come back to a Continuous trading phase as defined by its pattern at this time. Before coming back to Continuous an Uncrossing is performed.

When the Uncrossing is triggered if the uncrossing price lies within the collars the uncrossing is performed and a public **MarketStatusChange** (1005) message is disseminated to the market.

Right after the status change a public **PriceUpdate** (1003) message is sent to the market with the uncrossing price and the quantity at which the uncrossing is performed.

For each trade generated a public **MarketUpdate** (1001) is sent for the trade along with a public **OrderUpdate** (1002) message to update the corresponding orders.

At the end of the uncrossing process a public **MarketUpdate** (1001) message is sent to update the values of each limit that has changed.

When the uncrossing is fully performed the instrument switches to a Continuous trading phase. It is notified to the market by a public **MarketStatusChange** (1005) message.
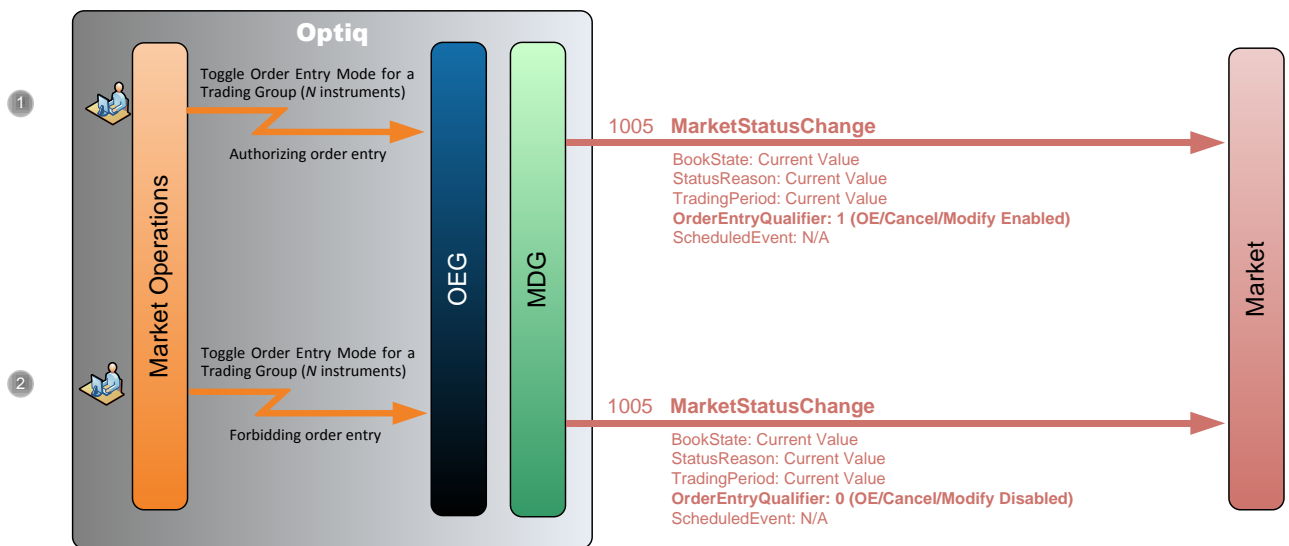
4.2.3   Instrument Reopened in Call phase
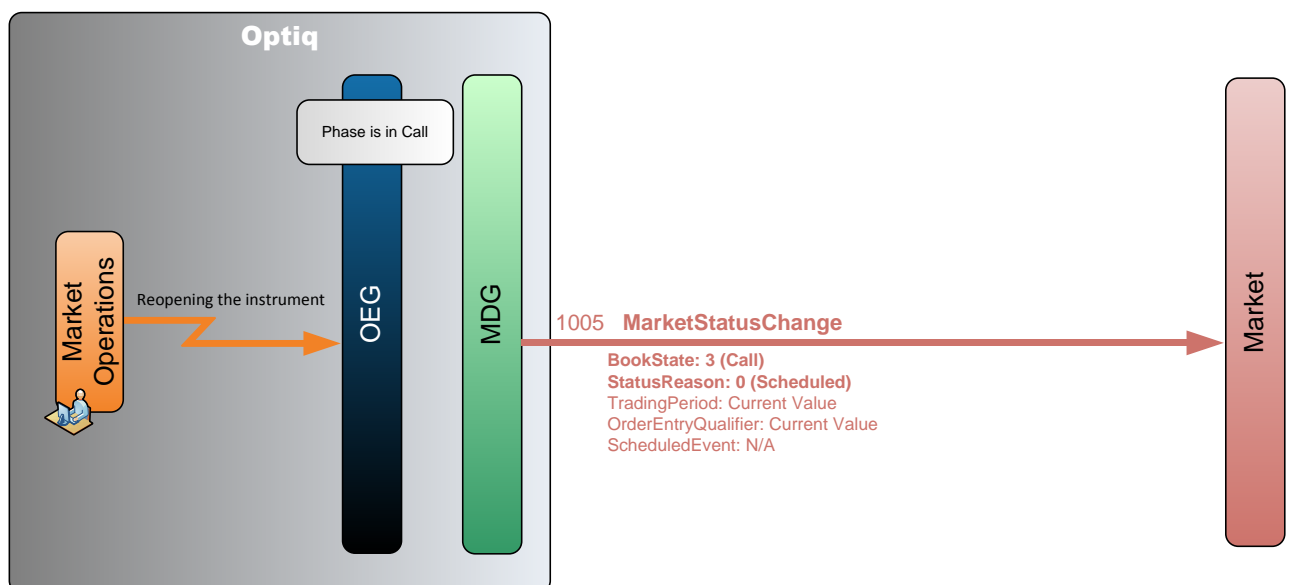
Market Operations reopens a specific instrument that was previously suspended.

Optiq sends a public **MarketStatusChange** (1005) message to the market to indicate that the instrument has resumed the original scheduled phase (Call in this case).

4.2.4   Market Operations Update Instrument Order Entry Mode for a Trading Group

①   Market Operations updates order entry authorization for a Trading Group.
A public **MarketStatusChange** (1005) message is sent to the market for each instrument of the
Trading Group to indicate that order entry is now authorized.

②   Market Operations updates order entry authorization for a Trading Group.
A public **MarketStatusChange** (1005) message is sent to the market for each instrument of the
Trading Group to indicate that order entry is now forbidden.
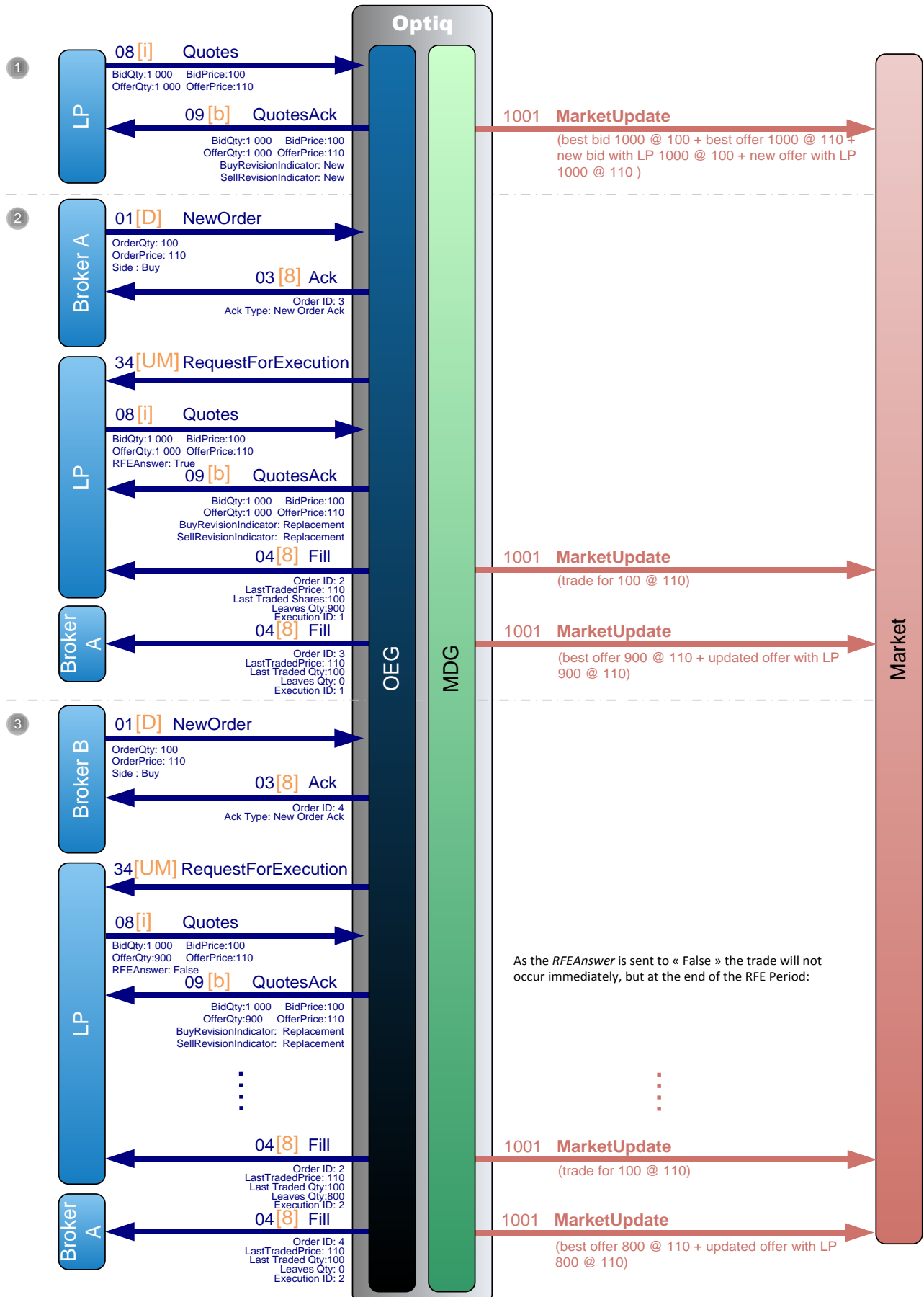
5.   WARRANT SPECIFIC MESSAGES

5.1   WARRANT COMMON KINEMATICS

5.1.1   Quotes message
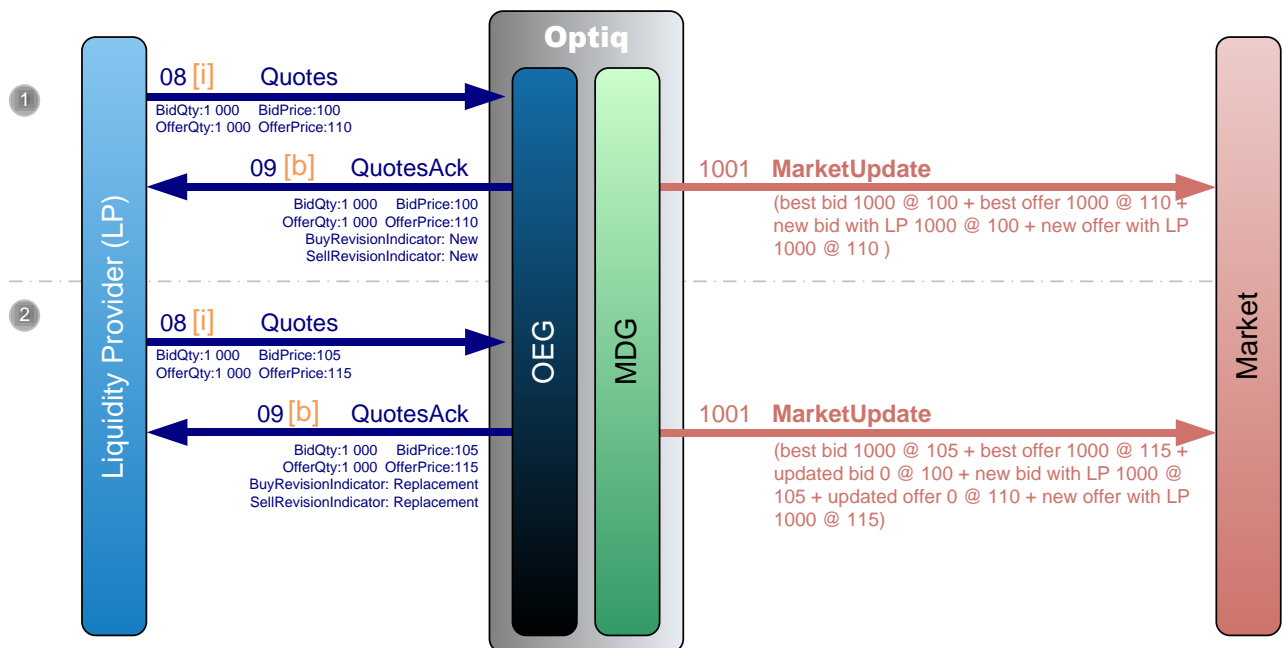
①   A LP sends a private **Quotes** (08) (FIX i) message to enter a new Buy quote with a quantity of 1,000 at a price of 100 along with another Sell quote with a quantity of 1,000 at a price of 110.

OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

The quotes enter the order book without matching and a public **MarketUpdate** (1001) message is sent to the market to update the limits and the BBO.

②   The same LP sends a private **Quotes** (08) (FIX i) message to revise his buy quote with a new price of 105 and his sell quote with a new price of 115.

OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes. The *Revision Indicator* for the both sides is set to 'Replacement' as these new quotes are replacing the old ones.

The quotes enter the order book without matching and a public **MarketUpdate** (1001) message is sent to the market to update the limits and the BBO.

5.1.2　Request For Execution

①   A LP sends a private **Quotes** (08) (FIX i) message to enter a new Buy quote with a quantity of 1,000 at a price of 100 along with another Sell quote with a quantity of 1,000 at a price of 110.

        OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

        The quotes enter the order book without matching and a public **MarketUpdate** (1001) message is sent to the market to update the limits and the BBO.

②   Broker A sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 100 and a price of 110.

        OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

        The order enters the order book and does not match, waiting for the end of the RFE period or the confirmation of the quotes by the LP. No public message is sent.

        OEG sends a private **RequestForExecution** (34) (FIX UM) message to the LP to let him the opportunity to confirm or modify the quote before completing the trade.

        The LP sends back a private **Quotes** (08) (FIX i) message to confirm his quotes; a Buy quote with a quantity of 1,000 and a price of 100 along with a Sell quote with a quantity of 1,000 and a price of 110 and the *RFE Indicator* set to 'True'.

        OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

        The quotes sent by the LP with the *RFE Answer* set to 'True' triggers the immediate execution of all the possible trades. The order immediately matches with the confirmed quotes and OEG sends back a private **Fill** (04) (FIX 8) message to the broker A and to the LP to publish the trade execution.

        A public **MarketUpdate** (1001) message is sent to the market for the trade followed by another **MarketUpdate** (1001) message to update the limits and the BBO.

③   Broker B sends a private **NewOrder** (01) (FIX D) message to enter a new Buy order with a quantity of 100 and a price of 110.

        OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the order.

        The order enters the order book and does not match, waiting for the end of the RFE period or the confirmation of the quotes by the LP. No public message is sent.

        OEG sends a private **RequestForExecution** (34) (FIX UM) message to the LP to let him the opportunity to confirm or modify the quote before completing the trade.

        The LP sends a private **Quotes** (08) (FIX i) message to confirm his quotes; a Buy quote with a quantity of 1,000 and a price of 100 along with a Sell quote with a quantity of 900 and a price of 110 and the *RFE Answer* set to 'False'.

        OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

The quotes sent by the LP with the *RFE Answer* set to 'False' will not trigger the immediate execution of the possible trades. No public message is sent.

At the end of the RFE period the order matches with the confirmed quotes and OEG sends back a private **Fill** (04) (FIX 8) message to the broker A and to the LP to publish the trade execution.

A public **MarketUpdate** (1001) message is sent to the market for the trade followed by another **MarketUpdate** (1001) message to update the limits and the BBO.

5.2   WARRANT SPECIFIC MARKET STATUS CHANGE

5.2.1   Beginning of a Bid Only Situation
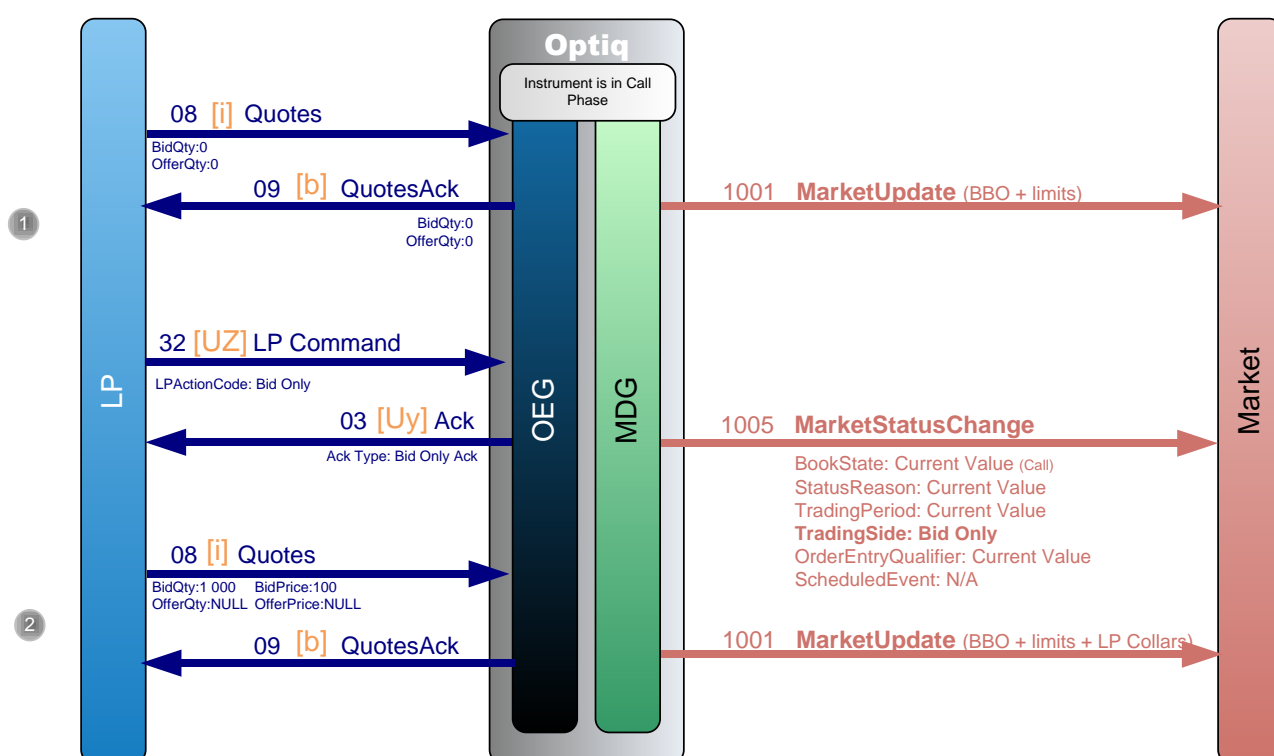
When a Liquidity Provider animating an instrument has no more instruments to sell, he moves to a Bid Only situation by entering the appropriate command. This command can be entered only if the Liquidity Provider has no more Quotes on the Instrument.

① The Liquidity Provider cancels the quotes and receives an Ack. The market is notified.

The Liquidity Provider submits a private **LiquidityProviderCommand** (32) (FIX UZ) message to change the instrument trading side from normal to Bid Only.
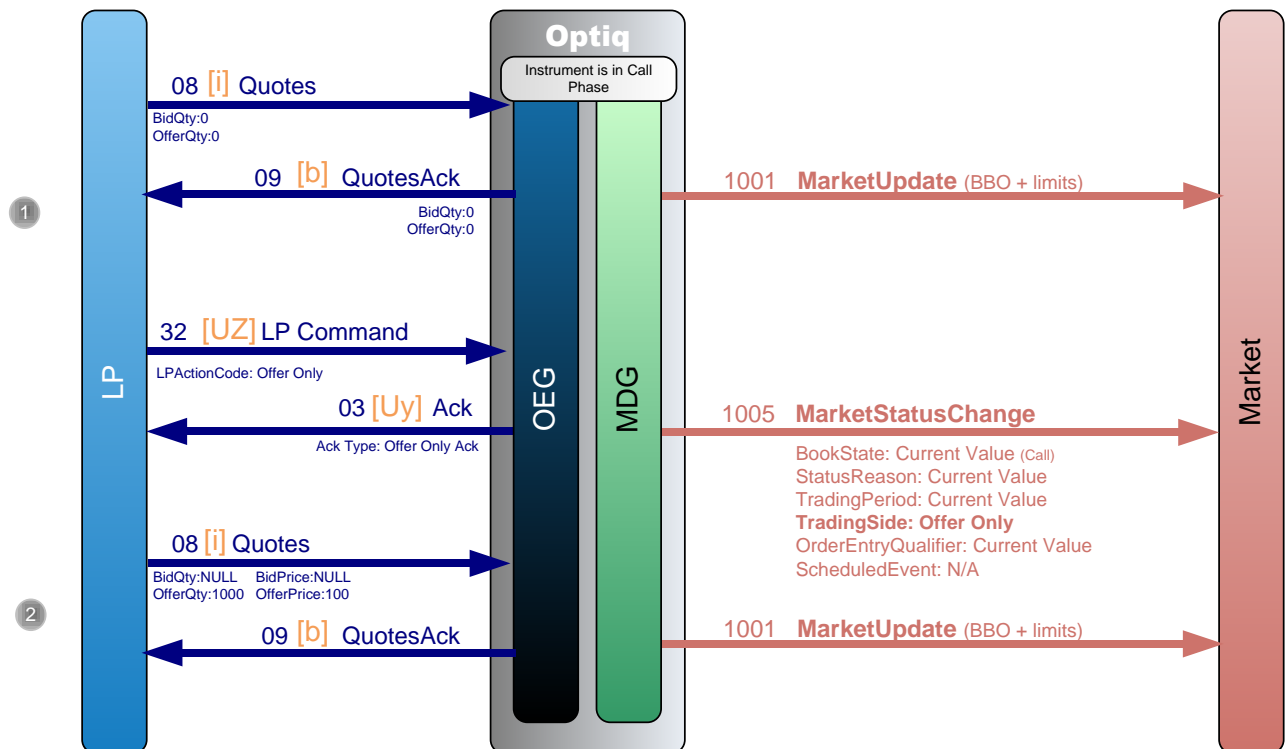
OEG sends back a private **Ack** (03) (FIX Uy) message to confirm the successful receipt and technical processing of the command.

A public **MarketStatusChange** (1005) message is sent to the market for the Bid Only trading period.

② After sending the **LiquidityProviderCommand** (32) (FIX UZ) message, LP needs to send the **Quotes** (08) message with the offer side values set to 'Null'.

 1.3.0

Note: LP collars are computed based on the bid quote and disseminated during Bid Only Situation.

5.2.2 Beginning of an Offer Only Situation

When a Liquidity Provider animating an instrument values his product at a bid price lower than one tick, he moves to an Offer Only situation by entering the appropriate command. This command can be entered only if the Liquidity Provider has no more Quotes on the Instrument.

① The Liquidity Provider cancels the quotes and receives a **Quotes Ack** (09) (FIX b) message. The market is notified.

The Liquidity Provider submits a private **LiquidityProviderCommand** (32) (FIX UZ) message to change the instrument trading side from normal to Offer Only.

OEG sends back a private **Ack** (03) (FIX Uy) message to confirm the successful receipt and technical processing of the command.

A public **MarketStatusChange** (1005) message is sent to the market for the Offer Only trading period.

② After sending the **LiquidityProviderCommand** (32) (FIX UZ) message, LP needs to send the **Quotes** (08) (FIX i) message with the bid side values set to Null.

         1.3.0

5.2.3　End of a One Side Only Situation (LP Quote Driven Warrant Market Model)
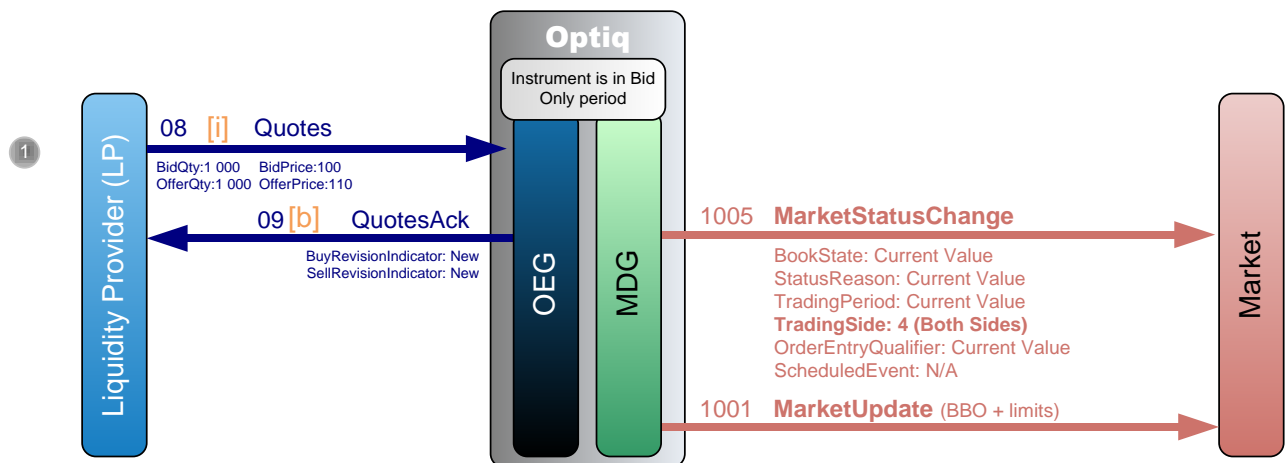

The instrument in in a Bid Only (or Offer Only) state.

① A LP sends a private **Quotes** (08) (FIX i) message to enter a new Buy quote with a quantity of 1,000 at a price of 100 along with another Sell quote with a quantity of 1,000 at a price of 110.

OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

The entry of a dual sided quote in a one side only period triggers the change of the Trading Side to the standard mode. Hence a public **MarketStatusChange** (1005) message is sent to notify the market the end of the Bid Only trading period.

Then the quotes enter the order book without matching and a public **MarketUpdate** (1001) message is sent to the market to update the limits and the BBO.

5.2.4   Knock-In by Issuer
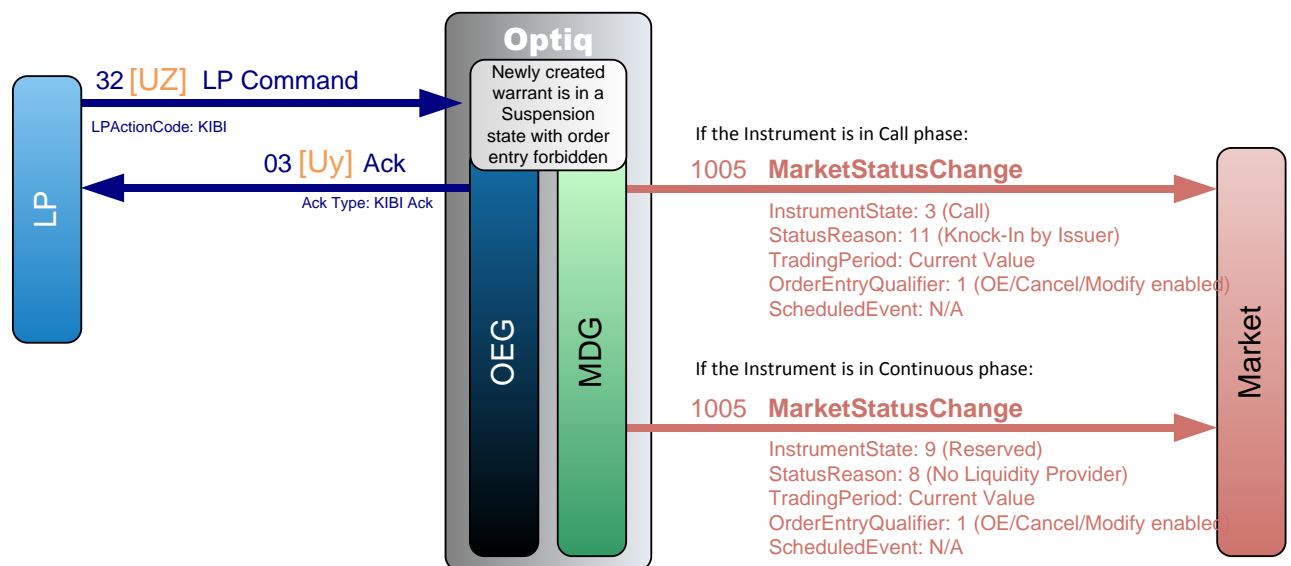
A Liquidity Provider has the ability to unsuspend a newly created warrant instrument by sending a private **LiquidityProviderCommand** (32) (FIX UZ) (Knock-In request). OEG sends back a private **Ack** (03) (FIX Uy) message to confirm the successful receipt and technical processing of the command.

If the command is sent during a Call phase, a public **MarketStatusChange** (1005) message is sent to inform the market that the warrant is unsuspended (with status reason 'Knock-In by Issuer').

If the command is sent during a Continuous phase, as the Liquidity Provider has not yet sent his quotes a public **MarketStatusChange** (1005) message is sent to the market as the warrant is unsuspended but reserved because there is no quote of the Liquidity Provider (status reason 'No Liquidity Provider').

Note: This functionality apply only to instruments that have not yet been opened to trading, otherwise the command is rejected.

5.2.5   Knock-Out by Issuer
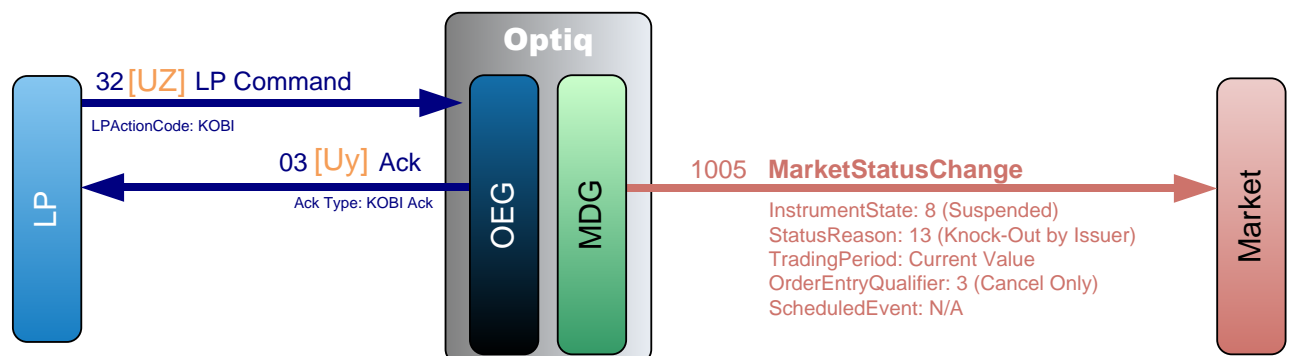
A Liquidity Provider has the ability to knock-out a warrant instrument.

Upon reception of a private **LiquidityProviderCommand** (32) (FIX UZ) (KOBI request), OEG sends back a private **Ack** (03) (FIX 8) message to confirm the successful receipt and technical processing of the command.

A public **MarketStatusChange** (1005) message is sent to inform the market that the warrant is suspended (with status reason 'Knock-Out by Issuer'). The order entry is set to Cancel Only (3) and LP quotes (if any) are automatically cancelled.

Note: The **LPCommand** (32) (FIX UZ) for KOBI requests only can be sent during the Closed trading phase.

5.2.6   Underlying Status Change
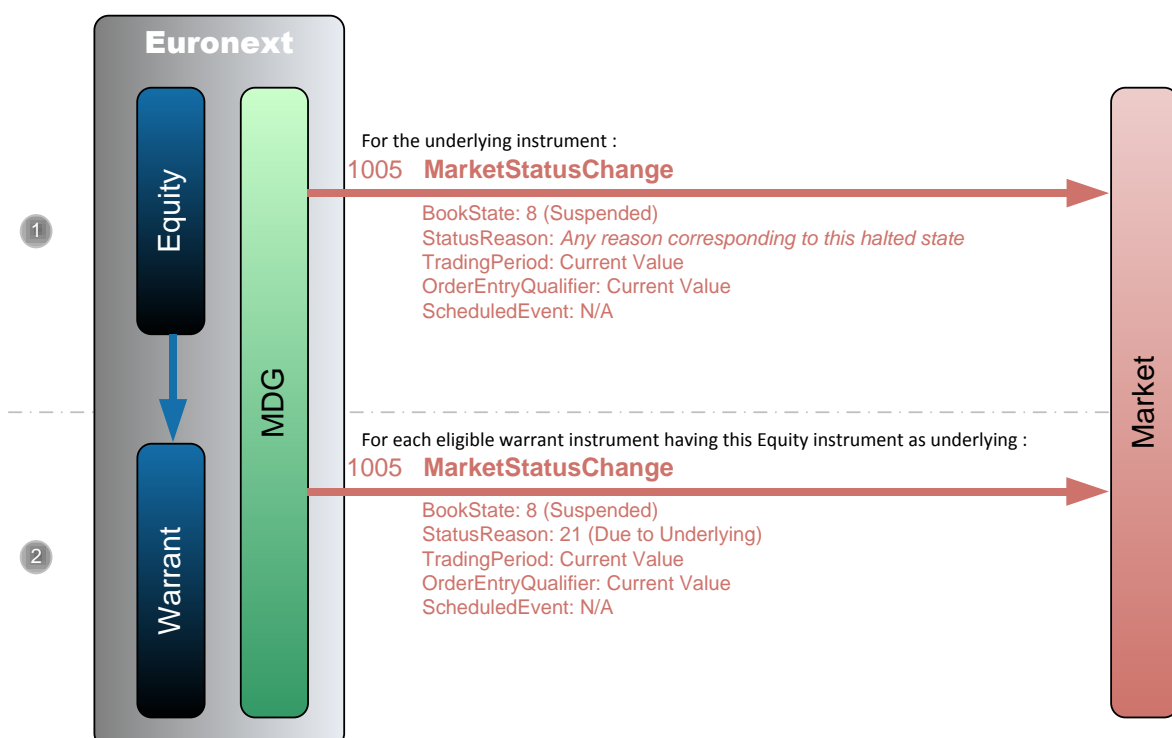
① The underlying instrument of a warrant instrument is suspended (for any reason) or suspended by Market Operations. A public **MarketStatusChange** (1005) message is sent to notify the market that the instrument has been reserved or suspended.

② Each configured eligible warrant having this instrument as underlying is accordingly suspended. A public **MarketStatusChange** (1005) message is sent for each instrument to notify the market that the instrument has been suspended.

Note: The Reopening underlying kinematic is similar to the suspension scenario: the status of the underlying affects the status of the warrant, i.e. Instrument State returns to its original scheduled phase and its Status Reason is 'Automatic Reopening'.

Exception to the reopening warrant kinematics: warrants that were suspended by Market Operations remain in that state.

5.2.7　Reserved No LP / LP is back
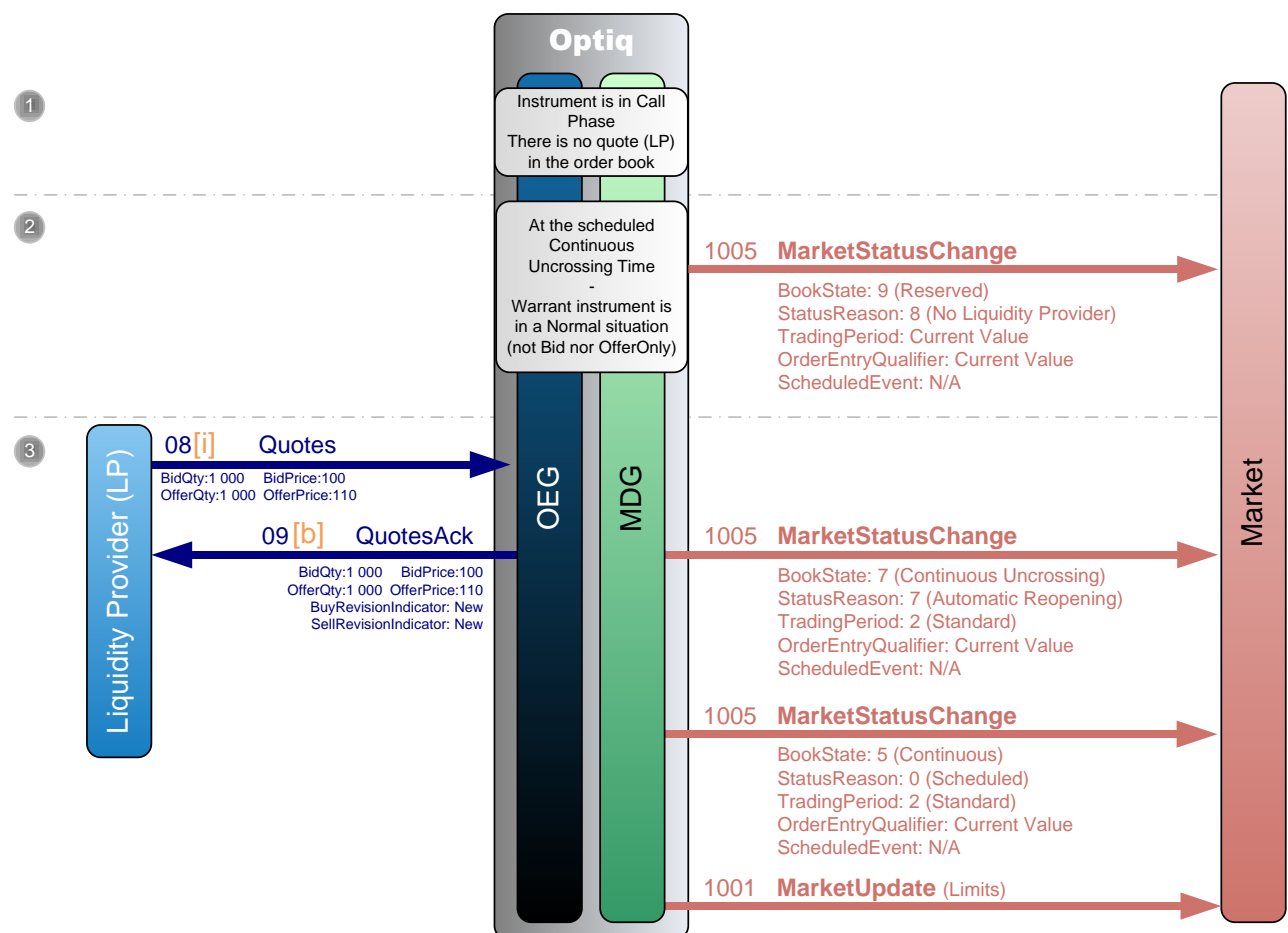

① In Call phase a warrant instruments is not reserved even if there is no quote from the Liquidity Provider in the order book.

②   At the Continuous Uncrossing time as there is no quote from the LP in the order book, the instrument is reserved and a public **MarketStatusChange** (1005) message is sent to notify the market that the instrument has been reserved.

③   A LP sends a private **Quotes** (08) (FIX i) message to enter a new Buy quote with a quantity of 1,000 at a price of 100 along with another Sell quote with a quantity of 1,000 at a price of 110.

OEG sends back a private **QuotesAck** (09) (FIX b) message to confirm the successful receipt and technical processing of the quotes.

As now there are quotes from the LP in the order book, the instrument is reopened and the book is uncrossed, a public **MarketStatusChange** (1005) message is sent for the reopening, and goes back to its scheduled phase. It is communicated to the market by a public **MarketStatusChange** (1005) message.

Then **MarketUpdate** (1001) message is sent to the market to update the limits.

5.2.8   Automatic Knock-Out by Exchange
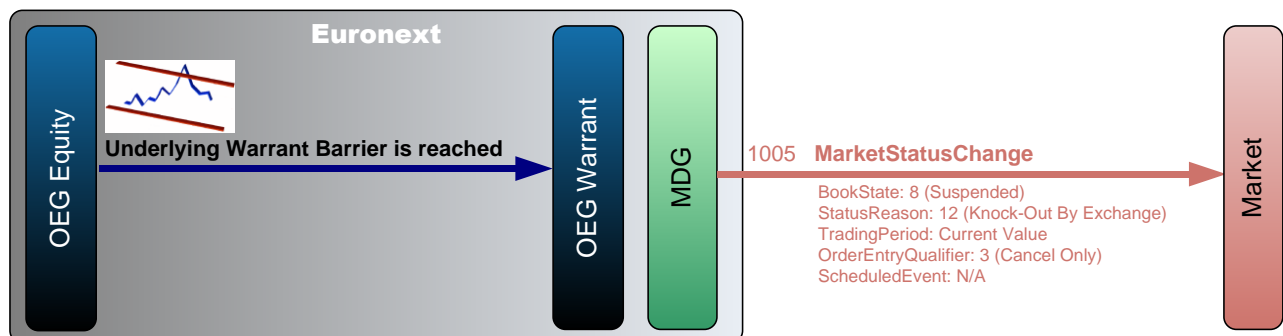
When the underlying price breaches the deactivation barrier of the warrant instrument, the warrant is automatically Knock-Out. A public **MarketStatusChange** (1005) message is sent to the market to communicate the suspension of the instrument (Status Reason 'Knock-Out by Exchange'). The order entry is set to Cancel Only and LP quotes (if any) are automatically cancelled.

5.2.9   Reactivating a Warrant (Quote Driven Warrant Market Model)
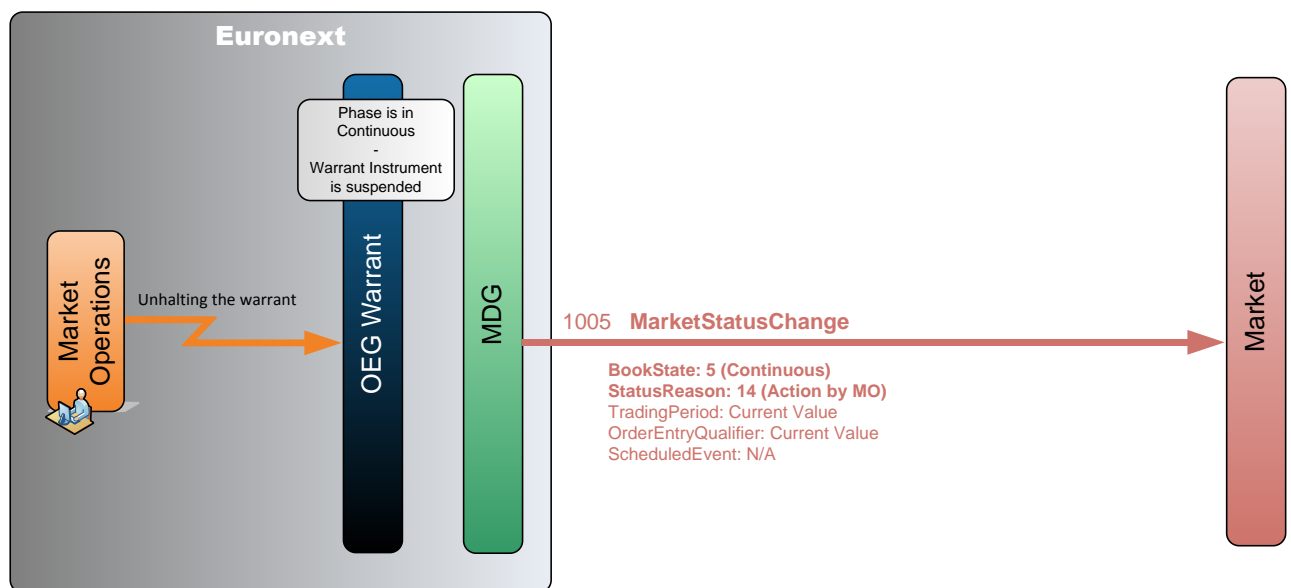
When a warrant erroneously reaches its deactivation barrier by the underlying instrument (e.g. deactivation barrier incorrectly entered at the warrant creation for example), the warrant is automatically suspended with order entry disabled. The warrant must be reauthorized for trading.

Market Operations reopens the warrant, authorizing it to trade again (according to the market phase). A public **MarketStatusChange** (1005) message is sent to inform the market that the warrant has been reopened by Market Operations.

5.2.10 Starting Payment After a Knock-Out Period
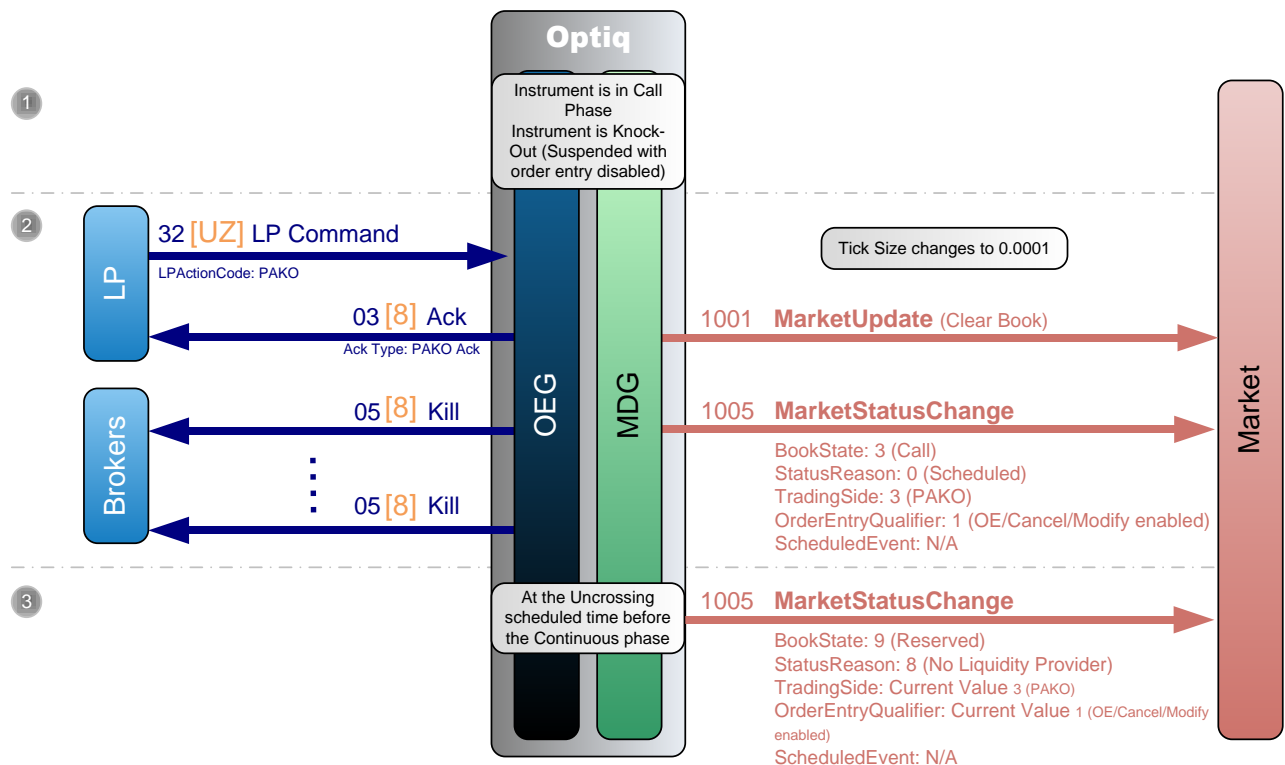
① An instrument has been knocked-out because trading on its underlying has breached the authorized trading barriers and is in a suspended state. The LP wants to start Payment After a Knock-Out period.

② The Liquidity Provider submits a private **LiquidityProviderCommand** (32) (FIX UZ) message with *Action Code* = PAKO, because he wants to trade back the residual value of this instrument after the knock-out.

   OEG sends back a private **Ack** (03) (FIX Uy) message to confirm the successful receipt and technical processing of the command.

   Before the beginning of the PAKO period, all resting orders in the order book are killed and the brokers are notified for their killed orders by **Kill** (05) (FIX 8) messages. A **MarketUpdate** (1001) message is sent to the market to clear the order book.

   Then a public **MarketStatusChange** (1005) message is sent to the market for the instrument state change and the beginning of the PAKO period.

③ At the scheduled Uncrossing time as there is no quote from the LP, the instrument is reserved. A public **MarketStatusChange** (1005) message is sent to notify the market that the instrument is reserved.

   The kinematic describing the return of the Liquidity Provider to the market is explained in section 5.2.7 Reserved No LP / LP is back.

Note: During the PAKO period, the *TradingSide* remains at '3' (PAKO) and the tick size is forced to 0.0001 but no additional message is sent to the market.

6.　REVISION HISTORY

Version	Change Description
1.3.0	**The following sections have been updated:** - 5.1.2 Request For Execution: corrected typo - 5.2.1 Beginning of a Bid Only Situation: added note for LP Collars dissemination - 2.3.8 Iceberg Order Refilled: updated note on iceberg randomisation
1.2.1	**The following global modifications have been performed:** - Correction of typos and replacement of references to "Euronext" with Optiq in kinematics diagrams - Correction of references to Halting with references to Suspension and Reservation - Correction of references to Class with references to Trading Group **The following sections have been updated:** - 1.2.5.1 Private and Public feed reconciliation: updated note on *Order Priority* definition - 5.2.3 End of a One Side Only Situation (LP Quote Driven Warrant Market Model): removed note on *Trading Side* **The following sections have been removed:** - 4.2.4 Class Auction (Class Level): it was a duplicate of section 4.1.1 Scheduled Uncrossing
1.2.0	**The following global modifications have been performed:** - Updated messages, descriptions, fields names and values according to Optiq OEG and MDG specifications - Added details in the descriptions to improve readability - Removed reference to message Extended Response - Removed reference to Systematic Internalizer (SI) - Addition of Inaccessible book state **The following sections have been added:** - 2.2.2.2 Logon Rejection in FIX - 2.2.3.2 Logout in FIX - 2.5.4 Cancel on Disconnect Mechanism - 2.8 Indicative Price Inputs - 2.8.1 Valuation Trade by Liquidity Provider Price Input message - 4.2.1 Market Operations Update Instrument Order Entry Mode for a Specific Class (Class Level) - 4.2.4 Instrument Unhalted in Call phase (Instrument Level) - 5.2.9 Reactivating a Warrant (Quote Driven Warrant Market Model Kinematic) (Instrument Level) **The following sections have been updated:** - Associated documents: Added reference to OEG FIX 5.0 specifications - 1.2.1 Private Messages: Updated messages list table (removed unused messages) - 1.2.2 Public Messages: Updated messages list table (removed unused messages) - 1.2.5.1 Private and Public feed reconciliation: Added clarification on the Order Priority field - 2. Common Kinematics: Added note on repeating sections use in messages - 2.1.1 Initialisation of a New Trading Day: Update graph and description with Inaccessible book state - 2.1.2 End Of Day: Update graph and description with Inaccessible book state - 2.2 Admin Messages: Renamed from ꞌAdmin Messages SBEꞌ to ꞌAdmin Messagesꞌ - 2.2.1 Successful Logon: Update graph and description with FIX specific details

Version	Change Description
	- 2.2.2 Logon Rejection: Added global logon description, updated SBE graph and description, added FIX dedicated graph and description
	- 2.2.3 Logout: Added global logon description, updated SBE graph and description, added FIX dedicated graph and description
	- 2.3.4 New Order Rejected: Updated FIX message reference in description
	- 2.4.1 Modifying an Unmatched Order: Side of the second order corrected from 'Buy' to 'Sell' in kinematics
	- 2.6.1 Ownership request for a specified order ID: Updated graph and description following the removal of Extended Response message, added kinematics for Drop Copy message sending
	- 2.6.2 Ownership request for a Logical Access: Updated graph and description following the removal of Extended Response message, added kinematics for Drop Copy message sending
	- 2.7.1 Market to Limit on Opening: Updated graph and description according to the latest message kinematics
	- 3.1.2 Automatic IMP Calculation: Updated note with indicative change in quantity case
	- 3.2.1 Reference Price Update: Updated graph and description with latest message kinematics
	- 3.2.3 Trade Cancellation: Removed PriceUpdate message sent after the trade cancellation
	- 4. Market Status Changes: Updated graph with latest repeating sections field values, added example in description
	- 4.2.2 Instrument suspended by Market Operations: Added note in description
	- 5.2.1 Beginning of a Bid Only Situation: Update description on FIX messages kinematics
	- 5.2.2 Beginning of an Offer Only Situation: Update description on FIX messages kinematics
	- 5.2.4 Knock-In by Issuer: Update section title
	- 5.2.5 Knock-Out by Issuer: Update section title
	- 5.2.10 Starting Payment After a Knock-Out Period: Update description on tick size changes and FIX ack
	The following sections have been removed: - 3.1.1 News and Announcements
1.1.0	**The following global modifications have been performed:** - Updated fields names and values according to Optiq OEG and MDG specifications - Added details in the descriptions to improve readability **The following sections have been added:** - 2.6 OWNERSHIP REQUEST - 2.6.1 Ownership request for a specified order ID - 2.6.2 Ownership request for a Logical Access **The following sections have been updated:** - 1.2.5.1 Private and Public feed reconciliation - 2.1.1 Initialisation of a New Trading Day - 2.1.2 End Of Day - 2.3.8 Iceberg Order Refilled - 2.3.11 Breaching a Collar with Confirmation (No Halt)
1.0.0	**First Version**

DOCUMENT HISTORY

REVISION NO.	DATE	AUTHOR	CHANGE DESCRIPTION
1.3.0	31 Jan 2018	IT Solutions – BA team – LPI	Minor updates
1.2.1	17 Nov 2017	IT Solutions – BA team – LPI DCO	Minor update on third version
1.2.0	16 Oct 2017	IT Solutions – BA team – LPI DCO	Third Version
1.1.0	16 Mar 2017	IT Solutions – BA team – LPI BSA AVE	Second Version
1.0.0	27 Oct 2016	IT Solutions – BA team – LPI FLO	First Version