BM&FBOVESPA
The New Exchange

EntryPoint: Order Entry Messaging

Messaging Guidelines

- Equities
- Derivatives
- FX

Version: 2.7
Last modified: 10/19/2015

Contacts

- Services Development Department (GDS): handles all requests for connectivity setup and general exchange supported services.
    - bvmfsolution@bvmf.com.br
- Certification and Testing Center (CTC): performs certification of all software solutions applying for EntryPoint connectivity.
    - ctc@bvmf.com.br
- Trading Support Department (GSN): provides real time connectivity monitoring and troubleshooting.
    - tradingsupport@bvmf.com.br
    - +55 11 2565-5000 (option 2)

Index

Figure Index

Change log

Date	Version	Description	Author
February 22nd, 2010	1.0	- Initial version.	AG
April 4th, 2010	1.1	- Various changes in order to accommodate for the backend trading system changes.	AG
September 20th, 2010	1.2	- Harmonized with the Unified Trading Platform.	AG
January 14th, 2011	1.2.1	- Table in Appendix E: updated.	EP
August 25th, 2011	1.3	- Describing Authentication, Session Connection and Cancel on Disconnect. - Added Advanced Functionalities section to describe User-Defined Spreads, Options Exercise and Forward.	AG EP
September 29th, 2011	1.3.1	- Updates in Authentication section.	EP
November 7th, 2011	1.4	- Describing Self-Trading prevention at customer level. - Added Deprecated Fields section.	EP
November 29th, 2011	1.5	- Added BEI (Brazil Easy Investing) section. - Updated NTP references to PUMA.	MARS EP
February 14th, 2012	1.6	- Added information about Aggressor Indicator. - Updated UDS diagrams. - Updated table of Validity Types Availability. - Added “Forward + Registered Cash” (TVR) Scenario. - Updated FX Market rules. - Updated Tables in Appendix E: and Appendix F:	EP
March 1st, 2012	1.6.1	- Removed reference to tag OrderRestrictions. - Updated list of PartyRole domain values. - Note about PositionEffect field usage. - Note about ThresholdAmount field usage.	EP
April 13th, 2012	1.6.2	- Describing Message Replay service. - Reviewed functionalities’ timeline. - Fixed example in section 6.7.1.3 - Added Appendix G:	EP
April 19th, 2012	1.6.3	- Fixed example in section 6.7.1.3	EP
May 7th, 2012	1.6.4	- Equalized ClOrdID/OrigClOrdID chaining rules in both Equities and Derivatives segments.	EP
May 23rd, 2012	1.7	- Removed Forward Declaration Cancellation by QuoteRequestID. - Updated timeframe availability of order validity “At the Close”. - Added warning about reuse of ClOrdID. - Described use of PartyRole 76 (Desk ID). - Described use of tag Memo.	EP
June 18th, 2012	1.7.2	- Updated Contacts information. - Added note about DMA2 in section 10.	EP

August 27th, 2012	1.7.3	- Documented change in the behavior of SecondaryOrderID and OrigClOrdID in section 6.7.3. - Updated scenario where a “Market with Leftover as Limit” order is filled twice.	EP
September 25th, 2012	1.7.4	- Updated Bovespa Numeric Range for DMA3 and DMA4 in section 10. - Updated FIX session identification table in section 4.1 - DMA participants can only initiate Forward deals. - Fixed diagrams in sections 15.1.2 and 15.1.3	EP
February 8th , 2013	1.7.5	- Described the throttle mechanism in section 3.7 - Updated order modification description in section 6.5 - Updated Rapid Firing restriction in section 6.7.1.4 - Updated tables in Appendix E: and Appendix F: to inform that orders with validity “Fill or Kill” (FOK) cannot be sent during Auction.	EP
February 18th , 2013	1.7.6	- Described Drop Copy in section 3.8	EP
March 4th , 2013	1.7.7	- Changed SenderLocation value for Give-up Agents in section 10.	EP
July 03rd , 2013	1.7.8	- OrderID is now guaranteed to be globally unique across all parameters. - Updated Rapid Firing restriction in section 6.7.1.4	EP
July 30rd , 2013	1.7.9	- Documented max length of PartyRole 76 (Desk ID) in section 12.2. - Warned about the use of Entering Traders by DMA participants that might coincide with desk trader IDs in section 8.	EP
November 25th, 2013	1.8	- Described New Order Cross scenario in section 15.1.6. - Described Average Price in section 7.2. - Added Trading Platform schedule in section 2.2. - Customers advised to disconnect during weekend in section 2.2.	EP
April 7th, 2014	1.9	- Fixed reference to values on tag OrderCategory in section 15.3.1 - Added Instrument Status identification in section 7.4	EP
September 1st, 2014	2.0	- Changed trade give-up process in section 13.2.2. - Described Market Protections functionality in section 14.7.	EP
October 10th, 2014	2.1	- Described Admin Session in section 3.9. - Removed tag OrdStatus (39) = 9 – Suspended in Appendix B:	EP / JLRM
October 21st, 2014	2.2	- Spelling corrections	JLRM
December 23rd, 2014	2.3	- Described, in section 3.9, the rules cancelation requests sent on Admin Session must conform to.	EP
March 16th, 2015	2.4	- Revised and updated. - Added examples for the different types of Market Protections in section 14.7. - Added scenarios of Market Protections functionality in section 15.9.	EP

July 13th, 2015	2.5	- Described Scheduled Exercise on Stock and Equity EFT's Options functionality in section 14.3. - Added scenarios of Scheduled Exercise on Stock and Equity EFT's Options in section 15.4. - Updated matrix in Appendix G:	EP
September 8th, 2015	2.6	- Updated Admin Session rules in section 3.9.	EP
October 19th, 2016	2.7	- Cancel On Disconnect functionality is available to Futures on Derivatives segment. - Removed BEI (Brazil Easy Investing) section.	EP

1. Preface

1.1 Description

EntryPoint is an evolution of BM&FBOVESPA previous order entry APIs and provides an improved, unified message specification allowing seamless access to multiple market segments, such as Equities, Fixed Income, Derivatives and Foreign Exchange.

1.2 Benefits

The unified API provides a number of advantages for the market participants:

- Less development effort, since most of the messaging behaviour stays the same across all markets.
- Single point of view for all markets, which eases development of multi market segment applications, such as cross market trading screens.
- Easier construction of inter market trading (e.g.: client side strategies)

1.3 System functions and characteristics

EntryPoint is based on the 4.4 version of the Financial Information eXchange ("FIX") Protocol. FIX is a technical specification for electronic communication of trade-related messages. It is an open standard managed by members of FIX Protocol Limited (http://www.fixprotocol.org/).

This document outlines the BM&FBOVESPA FIX implementation and is provided for third-parties which need trading connectivity through EntryPoint. It is assumed that the reader of this document has basic knowledge of the FIX protocol.

1.4 Intended Audience

This document outlines the BM&FBOVESPA FIX implementation and is provided for third-parties which need trading connectivity through EntryPoint[1].

1.5 Pre requisites

Not applied for this document.

1.6 Contract information

For EntryPoint contract information, please contact the BM&FBOVESPA's Business Service Development team through the email address bvmfsolution@bvmf.com.br or by phone +55 11 2565-7102.

For technical questions, please contact our Trading Support team through the email address tradingsupport@bvmf.com.br or by phone +55 11 2565-5000, option 2.

[1] The information within this document has been compiled by BM&FBOVESPA for general purposes only. BM&FBOVESPA assumes no responsibility for any errors or omissions. Additionally, all examples in this brochure are hypothetical situations, used for explanation purposes only, and should not be considered investment advice or the results of actual market experience.

All matters pertaining to rules and specifications herein are made subject to and are superseded by official BM&FBOVESPA rules. Current rules should be consulted in all cases concerning contract specifications.

2. Market Segments

BM&FBOVESPA products are structured in four market segments, namely: Derivatives and FX (former BM&F segment), Equities (former Bovespa segment), GLOBEX order routing, and Fixed Income (Brazilian government issued bonds).

The following table depicts the current product availability via the EntryPoint interface:

Market Segments			
Segment		Available via EntryPoint	Backend System
Derivatives (Former BM&F Segment)			
➔	Futures[2]	✓	PUMA
➔	Options2	✓	PUMA
➔	Forward	✓	PUMA
➔	Spot (Gold)	✓	PUMA
Foreign Exchange (FX)			
➔	BRL/USD cash market	✓	PUMA
Equities (Former Bovespa Segment)			
➔	Stocks	✓	PUMA
➔	Options on Stocks	✓	PUMA
➔	Forward on Stocks	✓	PUMA
➔	Exchange-Defined Strategies	✓	PUMA
➔	User-Defined Strategies	✓	PUMA
➔	Corporate issued bonds	✓	PUMA
GLOBEX order routing			
➔	CME Futures	✓	GLOBEX
➔	CBOT Futures	✓	GLOBEX
Fixed Income (FI)			
➔	Government bonds		Sisbex

2.1 Order Execution Rules at BM&FBOVESPA

BM&FBOVESPA matches orders by price/time priority. Lower offer prices take precedence over higher offers prices, and higher bid prices take precedence over lower bid prices. If there is more than one bid or offer at the same price level, earlier bids and offers take precedence over later bids and offers, respectively.

Under price/time priority of orders, a bid (offer) is filled at the best price by the earliest entered offer (bid) at that price. If additional contract units are needed to fill the bid (offer) then the next oldest offer (bid) at that price is matched until all of the liquidity at that price has been exhausted. Then matches would commence at the next best price until the order is completely filled.

[2] Encompass Financial Instruments (e.g. exchange/interest rates), Commodities (e.g. Corn, Soybean), and Indices (e.g. Bovespa Index).

2.2 Trading Platform Schedule

The following table describes the trading schedules for the platform in each market segment:

Segment	Schedule
PUMA Equities	Brought down daily between 22:00 and 3:00 (local time). On weekends between Fri 22:00 and Sun 12:00
PUMA Derivatives	Brought down daily between 22:00 and 3:00 (local time). On weekends between Fri 22:00 and Sun 12:00.

In general, for PUMA Trading System, customers may connect every day or keep connected through the week. BM&FBOVESPA highly recommends that customers remain disconnected during the weekends, unless when participating in scheduled mock tests.

3. Network Connectivity

The following sections describe all connectivity options for EntryPoint.

3.1 Physical/Link Layer Options

Market participants can choose from the following connectivity offers.

3.1.1 **RCCF**

RCCF ("Rede de Comunicações da Comunidade Financeira" or Financial Community Communications Network) is an MPLS network that connects all brokerage firms to BM&FBOVESPA, as well as some distributors and other interested clients.

This network allows for specific SLAs and contingency features. It is typically used to receive market data and transactional messages (order management).

3.1.2 **RCB**

RCB ("Rede de Comunicação BM&FBOVESPA" or BM&FBOVESPA Communications Network) is a newer communication option available to the BM&FBOVESPA customers.

Based on Ethernet over SONET (EoS/EoSDH), it allows participants to choose from a vast array of link speeds and service levels, which contrasts with RCCF, as the latter offers packaged, predefined solutions.

3.2 Network Setup for DMA Providers

DMA Providers may connect to BM&FBOVESPA to receive market data and to route orders to the BM&FBOVESPA broker community.

This connection may be established upon previous business agreement with BM&FBOVESPA, and the SLA is dependent on the type of information to be transferred over the network. The following diagram illustrates the possible setups for the network:

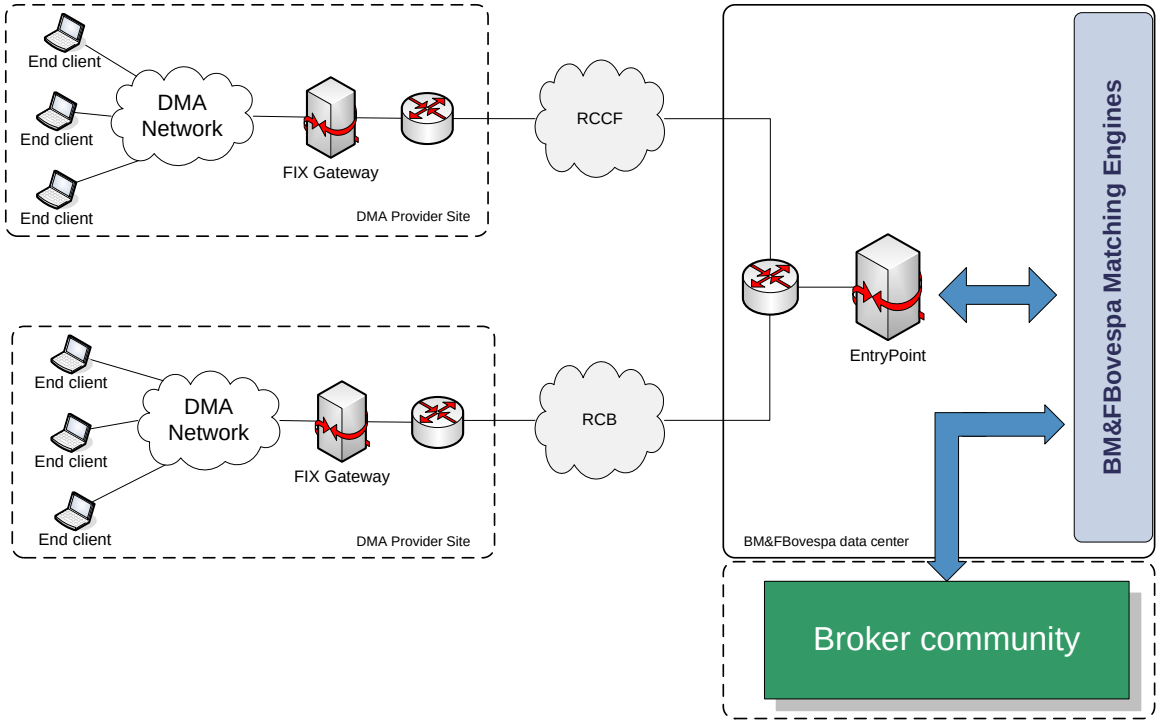

Figure 1 - BM&FBOVESPA DMA providers' network setup

3.3 Network Setup for Brokerage Firms/Banks

Brokerage firms and banks with trading rights at BM&FBOVESPA may connect to receive market data and orders routed from clients (for brokerage firms only), as well as issue their own orders according to previous business agreement with BM&FBOVESPA.

The following figure illustrates this setup:

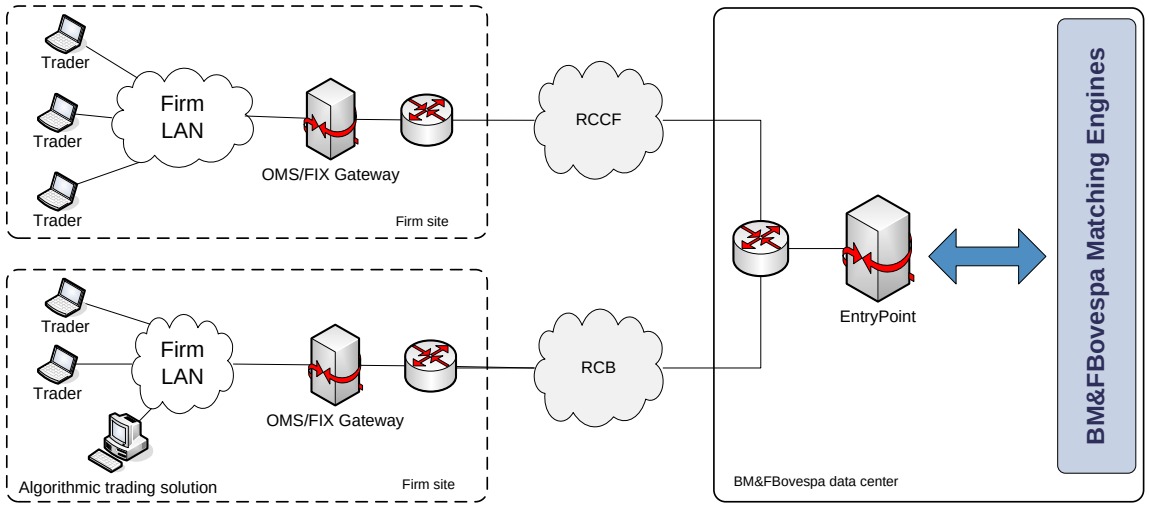

Figure 2 - BM&FBOVESPA-brokerage firms/banks network setup

3.4 Session Connection

All FIX session schedules for Entry Point are activated on a daily basis and, at the end of the session, the connection is terminated by the Border Gateway and the sequence number is reset. This means that at the beginning of every new session, the expected sequence number is 1.

Note that there will be no logout message when the session is terminated. Instead, users will detect a hard disconnection. Nevertheless, users may expect to receive all order expiration messages before the end of the daily session.

The time at which the Exchange starts and ends every session may vary according to the particular fix session. Please contact BM&FBOVESPA for more details on the start time and end time values defined for a particular session.

3.5 Authentication

FIX sessions may require the user to provide authentication data on the Logon message. The following table depicts the fields used to convey such information:

Tag	Tag name	Req'd	Data Type	Comment
95	RawDataLength	N	Length	Required when this message contains authentication data. Used to indicate the number of bytes in the RawData (95) field.
96	RawData	N	Data	Used to convey the password when this message contains authentication data.
925	NewPassword	N	String	Only sent from the client to BM&FBOVESPA. Allows the client to change its password.

3.5.1 Password Renewal

Passwords are initially provided by BM&FBOVESPA Trading Support Department (SSN) and then handed to clients to include in their applications. Users may change the password whenever it's necessary by sending a Logon message and providing the current password in tag RawData (96) and the new one in tag NewPassword (925).

3.5.2 Password Policy

In order to enforce security, some policies are in place and must be considered when changing passwords. The Password Policy comprehends the following aspects:

3.5.3 Password Age

By default, passwords are configured to not expire.

3.5.4 Session Lockout

In case a wrong password is provided, the authentication will fail and the connection will be shut down. A Logout message will be sent to indicate the failure in authentication. During the next 3 minutes the FIX session will be locked and no connection will be accepted within this time. The system allows up to 5 wrong attempts to establish a connection, after what the FIX session is locked and users will need to contact BM&FBOVESPA Trading Support Department (SSN) in order to restore the session.

3.5.5 Minimum Length

All passwords need to be at least 8 characters long. Requests for new passwords that don't conform to this requirement will be rejected.

3.5.6 Password History

The system records the last 10 passwords assigned to the FIX session. The new password must have not been used before.

3.5.7 Password Strength

In order to guarantee that passwords meet some strength requirements, the formation rule determines that all passwords must be composed of characters listed in three out of four categories:

- At least one lowercase character (from a through z)
- At least one uppercase character (from A through Z)
- At least one digit (from 0 through 9)
- At least one special character (non-alphanumeric)

Requests for new passwords that don't conform to this requirement will be rejected.

3.6 Cancel On Disconnect (COD)

Cancel on Disconnect provides the users of the electronic trading platform the option to have their orders automatically cancelled by the exchange during logout or when an abnormal disconnection occurs.

When the disconnection is detected, the system will attempt to cancel all non-GT orders, i.e. all orders with validity different of Good till Date (TimeInForce (59) = 6) and Good till Cancel (TimeInForce (59) = 1). Orders associated with the FIX Session entered on behalf will also be cancelled if COD is enabled for that FIX Session.

Once the connection is reestablished, user will be notified with Execution Report (35=8) messages indicating the cancelled orders, considering the customer is not making use of sequence reset upon reconnection.

In order to support Cancel on Disconnect, two optional fields have been added to the Logon (35=A) message layout: CancelOnDisconnectType (35002) and CODTimeoutWindow (35003). The next section depicts how those fields affect COD operation.

3.6.1 COD Type

In order to enable COD functionality, users must include the tag CancelOnDisconnectType (35002) on every Logon session message, indicating the criteria used to initiate it. If this tag is not present in the Logon message, then the default value is assumed and COD will not be enabled.

Tag	Tag name	Req'd	Data Type	Comment
35002	CancelOnDisconnectType	N	Char (1)	Criteria used to initiate COD by the Border Gateway. If this Tag is not present then COD will not be enabled. Valid values: 0 - Do Not Cancel On Disconnect Or Logout 1 - Cancel On Disconnect Only 2 - Cancel On Logout Only 3 - Cancel On Disconnect Or Logout

3.6.1.1 Do Not Cancel on Disconnect or Logout (default)

If tag CancelOnDisconnectType (35002) is not present in the logon message or if it's assigned 35002=0, COD functionality will not be enabled and no orders will be automatically cancelled.

3.6.1.2 Cancel on Disconnect Only

When users want their working orders to be automatically cancelled only upon an abrupt disconnection, tag CancelOnDisconnectType (35002) must be assigned with value 35002=1. In this case, when users gracefully terminate the session by sending a logout message, no orders will be cancelled.

3.6.1.3 Cancel on Logout Only

When users want their working orders to be cancelled only upon a logout, tag CancelOnDisconnectType (35002) must be assigned with value 35002=2. In this case, when an involuntary session disconnection occurs, no orders will be automatically cancelled.

3.6.1.4 Cancel on Disconnect and Logout

When users want their orders to be cancelled upon either a hard disconnection or an intentional logout, tag CancelOnDisconnectType (35002) must be assigned with value 35002=3. In this case, if by any means the session is terminated, the system will automatically try to cancel all applicable resting orders.

3.6.2 COD Timeout Window

The COD functionality was designed to provide a timeout window that allows the user to reconnect before the orders are cancelled. The value, in milliseconds, must be set in the Logon message using tag CODTimeoutWindow (35003). The countdown is started as soon as the loss of connectivity is detected. If this tag is not provided, it's assumed 35003=0.

Tag	Tag name	Req'd	Data Type	Comment
35003	CODTimeoutWindow	N	Int (5)	Border Gateway will not trigger COD if the customer reconnects within the timeout window (milliseconds) which starts when the triggering event is detected. Default if not specified is 0. Max allowed value is 60000.

The Border Gateway will not trigger Cancel on Disconnect if the connection is reestablished within the timeout window.

In case of a Border Gateway failure, the backup gateway will resume operations and immediately trigger COD for all eligible sessions (sessions that had COD enabled on the previous logon). Thus, in this scenario, the timeout window is not enforced.

3.6.3 COD Limitations

It's important to observe that COD functionality is intended to mitigate potential losses from unexpected disconnections. However, some conditions may prevent the correct operation of the feature and there's no guarantee that all resting orders will be successfully cancelled upon a disconnection.

For example, in case the market or the instrument is in a state that does not allow order cancelations, such as *Pre Close* and *Close* states, no orders will be cancelled by COD. Likewise, COD will not cancel an order that participates in an auction.

Also, it is important to indicate that COD will not be able to cancel an order that has been filled during the timeout window.

Upon a situation of disconnection, it's highly advised that firms contact market operations if there is uncertainty about the status of given orders.

The Cancel on Disconnect functionality is currently available for Equities and Futures.

3.7 Throttle

The throttling mechanism controls the flow of messages at the FIX session level and was implemented to regulate the number of messages sent to BM&FBOVESPA in order to optimize performance.

The throttling parameter is specified in messages per second and different actions may be taken should the throttle parameter be exceeded (queue or reject). So, two parameters may be set:

- The maximum amount of messages – which defines the maximum number of messages that will be processed per second;
- The reject / non-reject exceeded messages – defines if the exceeded messages must be rejected or queued.

If a message exceeds the maximum rate set, it can be rejected or queued. In case of rejection, a "Business Message Reject" error message will be sent with Business Reject Reason = “Throttle limit exceeded”. Client systems can cross-reference the business message reject message with the originating message that was throttled by verifying the content of tag 45 (RefSeqNum). This tag will contain the FIX session level sequence number (tag 34) of the message that was rejected.

If non-reject is set, the throttle mechanism will withhold the messages exceeded until the end of the second, in this case, a higher latency would be observed in the response.

Assuming a scenario in which the limit is set to 50 messages per second. The first period of time begins when the gateway receives the first message and if more than 50 messages are sent before the next second, they are throttled.

Throttling parameters are configured by the GSN (BM&FBOVESPA Trading Support Department) and are activated at the GSN’s discretion, or upon customer request.

3.8 Drop Copy

In order to provide client systems with the ability to manage risk in a near-time fashion, BM&FBOVESPA provides the drop copy functionality for the confirmation of orders, modification requests, cancel requests and fill notifications entered via the EntryPoint interface.

The drop copy functionality is made available through one or more FIX sessions, to which client systems may connect, and receive every order events.

Although all messages are currently sent to one specific FIX session, with the increase in message volume, drop copy sessions may be split into more FIX sessions to cater for better load balance and scalability.

Whereas current implementation caters for only one drop copy FIX session for all products, as migration to PUMA occurs, client systems must adapt to connect to more than one drop copy session to receive all messages pertinent to its order flow.

Application level messages can only be received by client systems. Any attempt from client systems to send application level messages, such as orders (New Order Single), will be rejected with a session level reject (tag 35=3).

3.8.1 Network Connectivity

Network connectivity to drop copy can be made through dedicated RCCF VPNs, Internet VPN or the RCB. Drop Copy services are available through DNS lookups, provided alongside the session information.

3.8.2 Fault Tolerance and Disaster Recovery

Drop Copy gateways are located in a cluster in the primary site that geographically expands to the backup site. That being, in case of a Fault Tolerance event due to a software or hardware issue, the backup node of the cluster in the primary site will take over for the primary node, applying the appropriate FIX sequence numbers.

In case of a Disaster Recovery event, the Drop Copy gateways in the backup site will take over for the primary site, ensuring no action is required on client systems.

3.8.3 Start of Day Procedures

The FIX session sequence number is reset to one at the beginning of each calendar day. Note that if client systems try to log on with a sequence number different than one at the start of day, the logon request will be rejected. However, during the day, client systems must not reset the sequence number, or all messages from the start of day will be retransmitted.

3.9 Admin Session

Under request, BM&FBOVESPA can provide a segregated FIX session to participants, exclusively to perform order cancelations. The main objective of this session is to replace the browser-based order management tool for cancelations, leaving the former tool as a contingency option.

Due to its unique purpose, the Admin Session does not accept other types of messages except the Order Cancel Request (35=F) and, in order to be fully processed, the request must adhere to specific rules.

First of all, the cancel request must provide the original order identifiers, such as the OrigClOrdID (tag 41) and its unique Order ID (tag 37).

Secondly, participants must use tag PartyRole (452) = 1001 to inform the original FIX session used to enter the order.

To finish, the content of tag Sender Location - tag PartyRole (452) = 54 - provided in the cancel request must always be “BVMF” and the Entering Trader - tag PartyRole (452) = 36 - must indicate the current admin session's SenderCompID used to send the Cancel request.

The following table illustrates an example of a DMA provider (e.g. “XYZ”) sending an Order Cancel Request (35=F) to BM&FBOVESPA, on an Admin Session, to cancel order “ABC1” originally sent on session “CZZZ123A”:

<table>
<tr><th>Session</th><th>Msg Sent</th><th>Msg Rec</th><th>OrderID</th><th>Orig ClOrdId</th><th>NoPartyIDs repeating group</th><th>Comments</th></tr>
<tr><td>ADMIN</td><td>F</td><td>--</td><td>123</td><td>ABC1</td><td>NoPartyIDs = 4
<table><tr><th>PartyID</th><th>PartyIDSource</th><th>PartyRole</th></tr><tr><td>BVMF</td><td>D</td><td>54 (Sender Location)</td></tr><tr><td>ZZZ</td><td>D</td><td>7 (Entering Firm)</td></tr><tr><td>OZZZ0001</td><td>D</td><td>36 (Entering trader)</td></tr><tr><td>CZZZ123A</td><td>D</td><td>1001 (Order Origination Session)</td></tr></table></td><td>Order Cancel Request sent on session OZZZ0001.</td></tr>
<tr><td>BROKER</td><td>--</td><td>8</td><td>123</td><td>ABC1</td><td>NoPartyIDs = 3
<table><tr><th>PartyID</th><th>PartyIDSource</th><th>PartyRole</th></tr><tr><td>XYZ</td><td>D</td><td>54 (Sender Location)</td></tr><tr><td>ZZZ</td><td>D</td><td>7 (Entering Firm)</td></tr><tr><td>OZZZ0001</td><td>D</td><td>36 (Entering trader)</td></tr></table></td><td>Order is cancelled.</td></tr>
</table>

As seen in the example above, the Execution Report message, confirming the order cancelation, is only received by the session that originally submitted the order. Note that the notification can be also captured in the Drop Copy session, but it is not received in the Admin Session.

4. FIX Messaging Guidelines

The following sections describe various messaging rules and aspects of FIX when applied to EntryPoint.

4.1 Session Identification

FIX connections are established based on “Comp IDs” – fields that identify, on the session level, the counterparty in the connection. These IDs do not convey trader or firm information. They are used only on the FIX session level. As per the FIX specification:

	SenderCompID	TargetCompID
A sends directly to B	A	B
B sends directly to A	B	A

The exchange assigned CompIDs follow the formations rules below (FFF, stands for firm mnemonic, XXXX for an alpha-numeric suffix):

Entity	SenderCompID Format
Trading Desk/Locals via exchange provided frontend (e.g. Megabolsa Station)	SFFFXXXX
Trading Desk/Locals via ISV provided frontend	CFFFXXXX
Give Up Agents	RFFFXXXX
DMA1 Connection	CFFFXXXX
DMA2 Connection	PFFFXXXX
DMA3 Connection	YFFFXXXX
DMA4 Connection	XFFFXXXX
Drop Copy: broker connection	DFFFXXXX
Drop Copy: client connection	ZFFFXXXX
Message Replay: broker connection	OFFFXXXX
Message Replay: client connection	LFFFXXXX

FIX Comp IDs and IP addresses for connection are assigned by BM&FBOVESPA to connecting counterparties. The process is differentiated according to the counterparty category (banks, trading firms, DMA providers, vendors, other exchanges). For more details, please contact the BM&FBOVESPA Trading Support Department (GSN).

4.1.1 Data Encryption

BM&FBOVESPA does not support built-in FIX encryption. Security of the connection is provided by the lower layers (MPLS in the case of RCCF, EoSDH in the case of RCB). BM&FBOVESPA market surveillance also performs audits in case there is the suspicion of illegal connection procedures.

4.2 Extensions to the official FIX specification

Despite of the effort to make EntryPoint as adherent to the FIX.4.4 specification as possible, in some scenarios it is necessary to extend and slightly alter the standard specification. Thus, the following might take place:

- Field addition: when a given message does not contain an appropriate field to convey a given business information, EntryPoint will derive from the official FIX specification. This might be in the form of adding user defined fields (tags 5000-9999, 20000-39999), fields that belong to the spec but are not part of the message, and imported fields from newer fix versions (e.g., using a FIX.5.0 field in FIX.4.4).

- Domain changes: when necessary, field domains can be expanded to account for unpredicted necessities. If the needed values are available in a newer FIX specification, preference is given to import these values into the EntryPoint domain. Domains are naturally reduced to the actual values supported. Thus, market participants should not expect to use all values available in the FIX specification.

- Message addition: as newer FIX specifications arrive, EntryPoint might draw from them and include newer official messages as opposed to creating user defined messages.

- Order of fields: users should not rely on receiving the fields inside a repeating group in the same order as presented in the specification, with exception of the delimiter, which will be always presented as the first item within the repeating group.

4.3 Data Restrictions

The FIX specification provides few restrictions regarding data types. However, an actual implementation must impose additional restrictions in order to guarantee correct operation. Thus, please observe the following:

- Maximum length: EntryPoint defines maximum field lengths for all fields in all messages. This length should not be overrun, even if that seems to work during test or certification. As the order data flows across different systems it is subject to different size limitations. The exchange recommended maximum size accounts for those differences, and guarantees the correct behavior. Keep in mind that the maximum size for a given field may vary across several messages.

- Field echoing: due to performance purposes, there is no automatic field echoing provisioning. This means that fields that are part of the official FIX specification but not of EntryPoint are not echoed in reply messages. This is especially relevant to participants that connect to multiple venues through the same application, which might end up sending official tags that will not be echoed.

- Domain restrictions: all EntryPoint specified fields are subject to domain restrictions. Thus, participants can only send domain values that are explicitly present in the specification. Unmentioned domain values may cause the message to be rejected by the means of the appropriate business reject message or a Session Level Reject (35=3) message.

4.4 Fields Deprecation

Messages sent by the Exchange, on Order Entry and Drop Copy sessions, used to include tag DeliverToCompID (128). However, the use of this field in any scenario is no longer supported as the Exchange has ceased providing this field.

Also, with the complete rollout of the PUMA Trading System on equities, tag ApplID (1180) is currently provided only on messages sent on the Drop Copy.

4.5 Best practices

In order to ensure correct operation of the FIX connectivity, please observe the following:

- Empty tags are not allowed and will be reject by the border gateways.
- Whitespace on string fields is allowed and has no special treatment. This means that the exchange systems might not trim inbound fields. Exchange provided identifiers (e.g. firm and trader codes, symbols, and account information) should be provided as assigned: for example, if the client account is '123', it should not be sent as '000123' or ' 123'.
- Do not code to side effects. For example, the exchange systems might order the message contents increasingly by the tag numbers. However, this behaviour is not mandatory by the FIX spec. Thus, avoid coding to undocumented behaviour, even if your application works as intended during certification. The application will be more resilient and require less maintenance.
- Avoid undocumented parsing: all exchange issued identifiers have published uniqueness rules, e.g.: OrderID is guaranteed unique across all orders on a given instrument and a specific trading date. In order to achieve the required uniqueness levels, exchange systems might concatenate several pieces of information such as date, symbol, and counter, among others. Client systems should not parse the tag components since the actual composition might change as the systems evolve. IDs must be treated as opaque identifiers which comply with a given uniqueness rule.
- Repeating group ordering: as per the FIX spec, the first repeating group field is always a repeating group counter. In each of the repetitions, the first field is the repeating group delimiter and thus is always required. Participants should not rely on additional rules such as numeric field sorting or ordering as the fields appear in the FIX specification.

- Repeating groups might be shuffled in outbound messages.

  Consider the message fragment:

...35=D…453=3|448=123|447=D|452=7|448=TRDR|447=D|452=36|448=BVMF|447=D|452=54|…		
Tag Name	Tag Number	Value
NoPartyIDs	453	3
PartyID	448	123
PartyIDSource	447	D
PartyRole	452	7 (Entering Firm)
PartyID	448	TRDR
PartyIDSource	447	D
PartyRole	452	36 (Entering Trader)
PartyID	448	BVMF
PartyIDSource	447	D
PartyRole	452	54 (Sender Location)

  The reply to this message might be as the following:

…35=8…453=3|448=123|447=D|452=7|448=BVMF|447=D|452=54|448=TRDR|447=D|452=36|…		
Tag Name	Tag Number	Value
NoPartyIDs	453	3
PartyID	448	123
PartyIDSource	447	D
PartyRole	452	7 (Entering Firm)
PartyID	448	BVMF
PartyIDSource	447	D
PartyRole	452	54 (Sender Location)
PartyID	448	TRDR
PartyIDSource	447	D
PartyRole	452	36 (Entering Trader)

  Although the FIX formation rules are still respected, the repeating group ordering has been changed. Client systems must be resilient to that.

5. Certification

BM&FBOVESPA has a certification environment used by the participants and by Independent Software Vendors (ISVs) for testing and certification purposes of their software before accessing the productive environment of the Exchange.

The validation and the tests on acquired or under development solutions can be carried out during work days from 9:00 to 19:00 (local time), with no follow up needed from the certification team.

For network connectivity to the exchange, please contact the Services Development Department (GDS).

The network setup is similar for all segments, since connectivity will be provided via a Certification FIX Gateway. The physical link used for certification may vary from the one to be used in production, since it is the application that is being certified, and not the physical layer. Hence, a client application which will run using RCCF or RCB in the production environment may be certified through an Internet VPN connection.

A certified solution of an ISV may be used by the market participants with no need to repeat the certification process. For customized solutions, the certification related to the tested software in the informed version is granted to the participant only.

The certification process comprises to run the script (test scenarios) for a product. After a technical analysis by the Certification and Testing Center (CTC), the participant will receive an e-mail from BM&FBOVESPA formalizing the certification and granting access to the production environment to the certified software.

When the participant considers that the application is ready for certification, he will connect to a certification FIX gateway, and will follow the BM&FBOVESPA certification document for order entry. The counterpart will perform a series of tests that are pre-specified by BM&FBOVESPA, and its outcome will be used as evidence of passing certification.

For certification arrangements, email the Certification and Testing Center (CTC). Check the ‘Contacts’ section for details.

Further details about the certification process are provided at BM&FBOVESPA's web site www.bmfbovespa.com.br follow ‘Services’ ➔ ‘Certification’.

6. Order Characteristics

The behavior of an order can be affected by many parameters, such as order type and validity. This section describes the types and modifiers which can be applied to a given order.

Although EntryPoint strives to maintain a consistent interface across all market segments, sometimes it is not possible to achieve this goal due to constraints such as underlying technology and market regulation. Thus, order concepts are described in the broadest way possible, and market specifics are clearly noted when appropriate.

6.1 Order Types

Order types are determined by the OrdType (40) tag. The following table depicts Order Type availability at the various segments.

Order Types	Market Segments			
	Equities	Fixed Income	Derivatives	FX
Market with Protection (40=1)	✓	✓		
Limit (40=2)	✓	✓	✓	✓
Stop with Protection (40=3)	✓	✓		
Stop Limit (40=4)	✓	✓	✓	
Market to Limit (40=K)	✓	✓	✓	

6.1.1 Market Orders with Protection (OrderType = 1)

Market orders (40 = 1) do not have an associated price. Upon entry, they “sweep” the order book, potentially generating multiple fills at different price levels until one of the following happens:

- The order is completely filled.
- The order reaches a *protection price level* (tag 35001 – ProtectionPrice field), which is automatically calculated by the matching engine. The protection price level represents the worst price that an order can fill. If there is still unfilled quantity and the next fill would occur at a price beyond protection, the market order rests as a limit order with price equal to the protection level price.

For bids, the protection price calculated is by adding an offset to the last trade price. For offers, the offset is subtracted from the last trade. The protection price cannot be specified in the incoming order.

The following tables depict how Market orders behave when the protection price is hit:

Starting Order Book (Last Trade Price: 10, Offset: 2 ➔ Protection Price (bid): 12)			
Bids		Offers	
Qty	Price	Price	Qty
		10	500
		11	300
		13	200

A buy market order for one thousand contracts is received. The following takes place:

Msg sent	Msg received	ClOrdId	OrdType	OrderID	Price	Qty	Protection Price	Exec Type	Ord Status	Last Qty	Last Price	Comment
D		ABC1	1		N/A	1000						
	8	ABC1	1	123	N/A	1000	12	New	New			Execution Report confirming receipt is sent back to customer.
	8	ABC1	1	123	N/A	1000	12	Trade	Partially Filled	500	10	Order is partially filled for 500 @ 10.
	8	ABC1	1	123	N/A	1000	12	Trade	Partially Filled	300	11	Second Execution: 300 @ 11.
The next price level (13) is higher than the protection price (12). Thus, the order will rest as a limited order at price 12. Subsequent Execution Reports will have OrdType = Limit.												

6.1.2 **Limit Orders (OrderType = 2)**

Limit orders (40 = 2) specify the worst price at which the order may execute, i.e. an order to buy a security at or below a stated price (defined in tag 44 – Price), or to sell a security at or above a stated price. If the order does not execute, it will remain in the order book.

6.1.3 **Stop Orders with Protection (OrderType = 3)**

Stop orders (40 = 3) have an associated trigger price but no limit price. Whenever a trade crosses the trigger price, the order is automatically inserted in the order book as a limit order. The order price is automatically determined by the trading engine by the addition or subtraction of an offset price in the same spirit of the Market Order with protection.

The following tables depict how Market orders behave when the protection price is hit:

Starting Order Book (Last Trade Price: 10, Offset: 2 ➔ Protection Price (bid): 12)			
Bids		Offers	
Qty	Price	Price	Qty
		10	500
		11	300
		13	200

A buy market order for one thousand contracts in received. The following takes place:

Msg sent	Msg received	ClOrd ID	Ord Type	Order ID	Price	Qty	Stop Price	Protection Price	Exec Type	Ord Status	Working Indictor	Last Qty	Last Price	Comment
D		ABC1	3		N/A	1000	10							
	8	ABC1	3	123	N/A	1000	10	12	New	New	N			Execution Report confirming receipt is sent back to customer.
A trade crossed (quantity = 100) the trigger price. Thus, the order will rest as a limited order at price 12. Subsequent Execution Reports will have OrdType = Limit.														

Msg sent	Msg received	ClOrd ID	Ord Type	Order ID	Price	Qty	Stop Price	Protection Price	Exec Type	Ord Status	Working Indictor	Last Qty	Last Price	Comment
	8	ABC1	2	123	12	1000		12	New	New	Y			Execution Report is sent back to customer.
	8	ABC1	2	123	12	1000		12	Trade	Partially Filled		400	10	Order is partially filled for 400 @ 10.
	8	ABC1	2	123	12	1000		12	Trade	Partially Filled		300	11	Second Execution: 300 @ 11.
The next price level (13) is higher than the protection price (12). Thus, the order will rest as a limited order at price 12. Subsequent Execution Reports will have OrdType = Limit.														

6.1.4 Stop Limit Orders (OrderType = 4)

Stop limit orders (40 = 4) are associated with a trigger price – or stop price – at which it becomes a limit order. The order will be inserted into the order book as soon as the first trade occurs at the specified stop price (tag 99 – StopPx). From this point on, it will behave as a regular limit order, with the price defined in tag 44 – Price. The Order Type behavior is similar to that of Market to Limit Orders: after triggering, it will explicitly change to a Limited Order (40 = 2).

In case of illiquid instruments, if a GT Stop Limit Order is accepted, the price variation might cause this order to be triggered outside of the hard limits. In such situation the order price will be adjusted according to the values of the intraday limits.

Example: If an Ibovespa buy Stop Limit Order is triggered at 70,200 points, but the high hard limit is 70089 points, the order will become a limit order at 70,085 points, which is an acceptable price according to instrument tick size (5 points for Ibovespa). This prevents off tick orders.

6.1.5 Market with Leftover as Limit (OrderType = K)

Market with leftover as limit orders (40 = K, also known as Market to limit) will behave like a regular market order, and any unexecuted quantity will become a limit order in the order book, with its limit price set at the last executed price. In a partial fill scenario, the OrdType (40) field is changed to '2' (Limit) in order to reflect the new order behavior.

Scenario: Market with Leftover as Limit order is filled twice

Msg sent	Msg received	ClOrdId	OrdType	OrderID	Price	Qty	Exec Type	Ord Status	Last Qty	Last Price	Comment
D		ABC1	K		N/A	7000					
	8	ABC1	K	123	N/A	7000	New	New			Execution Report confirming receipt is sent back to customer.
	8	ABC1	K	123	N/A	7000	Trade	Partially Filled	2000	10.58	Order is partially filled for 2000@10.58
	8	ABC1	2	123	10.58	7000	Trade	Partially Filled	1000	10.58	Second Execution: 1000@10.58

6.2 Order Validity Types (Time in Force)

The following table depicts Validity Type availability at the various segments.

Validity Type	Market Segments			
	Equities	Fixed Income	Derivatives	FX
Day	✓	✓	✓	✓
Immediate or Cancel	✓	✓	✓	
Fill or Kill	✓	✓	✓	
Good Till Date	✓	✓	✓	
Good Till Cancel	✓	✓	✓	
Good for Auction	✓	✓		
At the Close	✓	✓		

6.2.1 Day (TimeInForce = 0)

Day orders (59 = 0) are available in the order book during the day until they execute or are cancelled (either by the customer who submitted it or BM&FBOVESPA market operations). It is considered the default validity when none is specified.

In the Equities, Derivatives and FX segments, at the end of the day all orders are cancelled by the matching engine during the cancel orders trading phase, and the customers who submitted them will receive the Execution Reports expiring the orders (ExecType (150) = Expired (C)).

6.2.2 Good till Cancel (GTC) (TimeInForce = 1)

Good till cancel orders (59 = 1) never expire. They are inserted in the order book and remain until cancelation by the customer or market surveillance, or until it is fully executed. GTC orders are not restated to client systems at the start of every trading session.

In the Equities Segment, if a stock is affected by corporate actions (e.g. split, reverse split, dividend payment), all outstanding orders on that stock will be cancelled, even if the GTC validity is specified.

6.2.3 Immediate or Cancel (IOC) (TimeInForce = 3)

The Immediate or Cancel (IOC) validity (59 = 3), also known as Fill and Kill (FAK), indicates that the order requires immediate execution, and the unexecuted quantity is automatically cancelled. If there is no counterparty to execute against, the order is acknowledged then cancelled.

The following table depicts the scenario where the incoming order can be matched.

Msg sent	Msg received	ClOrdID	OrderID	Price	Qty	Exec Type	OrdStatus	Last Qty	Last Price	Comment
D		ABC1		10.58	7000					
	8	ABC1	123	10.58	7000	New	New			Execution Report confirming receipt is sent back to customer.
	8	ABC1	123	10.58	7000	Trade	Partially Filled	4000	10.58	Order is partially filled for 4000 @ 10.58.
	8	ABC1	123	10.58	7000	Cancelled	Cancelled			Order is cancelled.

6.2.4 Fill or Kill (FOK) (TimeInForce = 4)

Fill Or Kill (FOK) orders (59 = 4) require that the full amount stated in the order is executed upon entering the order book. If there is not enough quantity on the opposite side to fill the order, the order is acknowledged then cancelled. It is also known as All or Nothing (AON).

Msg sent	Msg received	ClOrdID	OrderID	Price	Qty	Exec Type	OrdStatus	Last Qty	Comment
D		ABC1		10.58	7000				Instrument is in continuous trading mode.
	8	ABC1	123	10.58	7000	New	New		Execution Report acknowledging the order.
	8	ABC1	123	10.58	7000	Cancelled	Cancelled		Execution Report cancelling the order, since there's no opposite side.

6.2.5 Good till Date (GTD) (TimeInForce = 6)

The Good till Date (59 = 6) validity causes the order to expire at the end of the trading session of the date stated in the ExpireDate (432) field of the original order submitted by the customer.

At the end of the trading session of the expiration date, the orders are cancelled by the matching engine and customers who submitted the orders receive Execution Reports expiring the orders (ExecType (150) = Expired (C)).

6.2.6 At the Close (MOC) (TimeInForce = 7)

The At the Close (59 = 7) validity allows the participants to place orders that participate in the closing auction in advance. For example, the order can be placed during continuous trading but will become active just when the closing auction starts. It is also known as Market On Close (MOC).

At the moment of placement, if the order is not rejected, participants will receive an acknowledgment (ExecType=New, OrdStatus=New) indicating that the order is accepted but not active (WorkingIndicator=N).

When the closing auction starts, an Execution Report is sent, notifying that the order is active (ExecType=New, OrdStatus=New, WorkingIndicator=Y).

The 'At the Close' validity can only be combined with Market Orders.

6.2.7 Good for Auction (MOA) (TimeInForce = A)

The Good for Auction validity (59=A) indicates that an order is valid for the ongoing auction only. It is also known as Market On Auction (MOA).

Thus, it can only be sent during an auction. Whenever the auction finishes, the order is expired.

The 'Good for Auction' validity can only be associated with Market Orders.

6.3 Order Quantities

EntryPoint supports a number of order quantities which can be used to accomplish determinate trading strategies, such as minimum guaranteed execution and partial order disclosure. Order quantities are discussed in the following sections.

6.3.1 Disclosed Quantity (Iceberg Orders)

Disclosed Quantity allows participants to trade a large lot of a given security without exposing the whole lot in the market at once. The MaxFloor (111) field determines the largest amount which is shown in the order book at a time. In example, an order with OrderQty = 10000 and MaxFloor = 500 will show in the order book as a 500 contract (shares) order. After the order is filled for 500 contracts, the matching engine will replenish the quantity back to 500 contracts, until all of OrderQty is consumed.

In the PUMA Trading System, in order to preserve the hidden nature of Iceberg orders, the matching engine will assign a new order identifier (SecondaryOrderID) each time the order is replenished.

Msg sent	Msg received	OrderID	Secondary OrderID	Qty	Max Floor	Last Qty	Leaves Qty	Ord Status	Exec Type
D				10000	500				
Iceberg order is accepted									
	8	ORD_1	ORD_1	10000	500		10000	New	New
Disclosed quantity of 500 shares is totally filled again									
	8	ORD_1	ORD_2	10000	500	200	9800	Partially Filled	Partial Fill
	8	ORD_1	ORD_2	10000	500	300	9500	Partially Filled	Partial Fill
Order is replenished and a new Order ID is sent									
	8	ORD_1	ORD_3	10000	500		9500	Partially Filled	**Restated**

Participants can disable the iceberg functionality by setting MaxFloor = 0 in the modification messages. This shows the whole remaining quantity in the order book.

6.3.2 Minimum Quantity

Orders with minimum quantity must execute at least the quantity stated in field MinQty (110) in every transaction. Orders whose minimum quantity may not be satisfied upon entry in the order book are cancelled.

Minimum quantity can also be specified on modification messages. The behavior is equivalent to the order entry scenario.

6.3.3 Trade Related Quantities

EntryPoint provides fields which can be used to track quantity state through executions. Those fields are sent in every Execution Report message:

- CumQty (14): indicates the accumulated quantity of all trades involving the order. Its value grows as subsequent fills take place. If a given trade for a fully filled order is cancelled (trade bust), the CumQty sent in the Execution Report message is set to zero. If a trade of a partially filled order is cancelled, the CumQty will decrease in the same value of the busted quantity. Participants can track the actual executed quantity by subtracting LastQty from the cumulative quantity maintained by the participant's application.

- LeavesQty (151): conveys the amount which is still open for execution. This quantity decreases with every fill. Please note that trade busts do not roll back the busted quantity. Thus, LeavesQty does not increase when a bust take place.

- LastQty (32): contains the traded quantity from the last execution. LastQty changes with every fill, and will stay the same between fills.

6.3.4 In-flight Modification and Interpretation of the OrderQty Field

The OrderQty field is interpreted by BM&FBOVESPA as the total investor quantity, i.e. the total size of the order. That stands true for order modification requests as well. Hence, the connecting counterparty must take this into consideration when implementing cancel/modification logic, especially regarding in-flight modification scenarios (where the order is executed at the exchange at the same time the counterparty issues a modification request for that same order).

The following scenarios represent a high-level overview of the messages exchanged (with the most relevant fields only) for different situations that represent the interpretation of OrderQty:

Scenario 1: Plain modification of previously sent order

In this scenario, an order is sent (BUY 1000 @ 12), and has its quantity increased to 1400 due to a modification request.

Msg sent	Msg received	ClOrdID	Orig ClOrdID	Order ID	Orig ClOrdID	Price	Qty	Ord Status	Exec Type	Cum Qty	Leaves Qty	Comment
D		ABC1				12.00	1000					New Order from trader.
	8	ABC1		ORD_1		12.00	1000	New	New	0	1000	Order is ack'ed by exchange.
Trader issues a modification request to increase quantity to 1400.												
G		MOD1	ABC1	ORD_1	ABC1	12.00	1400					Order qty is increased to 1400 by trader.
	8	MOD1	ABC1	ORD_1	ABC1	12.00	1400	Replaced	Replace	0	1400	Modification is ack'ed by exchange.

Scenario 2: In-flight modification of previously sent order

In this scenario, an order is sent (BUY 1000 @ 12), which partially executes for 200. Concurrently (i.e., before receiving the Execution Report notifying the partial fill of 200), the counterparty issues a modification request to increase its quantity to 1300.

Msg sent	Msg received	ClOrdID	Orig ClOrdID	OrderID	Price	Qty	OrdStatus	Exec Type	Cum Qty	Leaves Qty	Last Qty	Last Price	Comment
D		ABC1			12.00	1000							New Order from trader.
	8	ABC1		ORD_1	12.00	1000	New	New	0	1000			Order is ack'ed by exchange.
The order executes for 200 @ 12.00. The partial fill notification does not reach the trader yet. Trader issues a modification request for the same order, increasing quantity to 1300.													
G		MOD1	ABC1	ORD_1	12.00	1300							Order qty is increased to 1300 by trader.
Trader receives the Execution Report for the partial fill (200 @ 12.00)													
	8	ABC1		ORD_1	12.00	1000	Partially Filled	Trade	200	800	200	12.00	Partial fill is received by trader.
Here, the order is modified in the match engine, with its total investor quantity increased to 1300, even though 200 were already executed.													
	8	MOD1	ABC1	ORD_1	12.00	1300	Replaced	Replace	200	1100			Modification is ack'ed by exchange.

Scenario 3: In-flight modification of previously sent order

In this scenario, an order is sent (BUY 1000 @ 12.00), which partially executes for 800 (remaining quantity = 200). Concurrently, the counterparty issues a modification request trying to decrease the order quantity from the original 1000 to 700.

Msg sent	Msg Received	ClOrd ID	Orig ClOrdID	Order ID	Price	Qty	OrdStatus	Exec Type	Cum Qty	Leaves Qty	Last Qty	Last Price	Comment
D		ABC1			12.00	1000							New Order from trader.
	8	ABC1		ORD_1	12.00	1000	New	New	0	1000			Order is ack'ed by exchange.
The order executes for 800 @ 12.00. The partial fill notification does not reach the trader yet. Trader issues a modification request for the same order, decreasing quantity to 700.													
G		MOD1	ABC1	ORD_1	12.00	700							Order qty is decreased to 700 by trader.
Trader receives the Execution Report for the partial fill (800 @ 12.00).													
	8	ABC1		ORD_1	12.00	1000	Partially Filled	Trade	800	200	800	12.00	Partial fill is received by trader.
	8	MOD1	ABC1	ORD_1	12.00		Cancelled	Cancelled	800	0			Since the order is being modified so that OrderQty < CumQty, the order is cancelled.

6.4 Order Lifecycle

An order typically goes through a number of states during its lifecycle, such as new, replaced, cancelled, and filled, among others. Each of these states is conveyed by the OrdStatus (39) field sent in all Execution Report (35=8) messages.

When the participant performs an action to an order, such as modification or cancellation, the outcome of the operation is expressed in the ExecType (150) field. Thus, a successful modification report will have ExecType set to '5' (Replace).

Internal matching engine events, such as fills or end of day expiration might also lead to state changes.

The following Diagram depicts the basic state transitions common to all of EntryPoint market segments. Please refer to **Appendix B**: for a detailed list of ExecType and OrdStatus transitions.

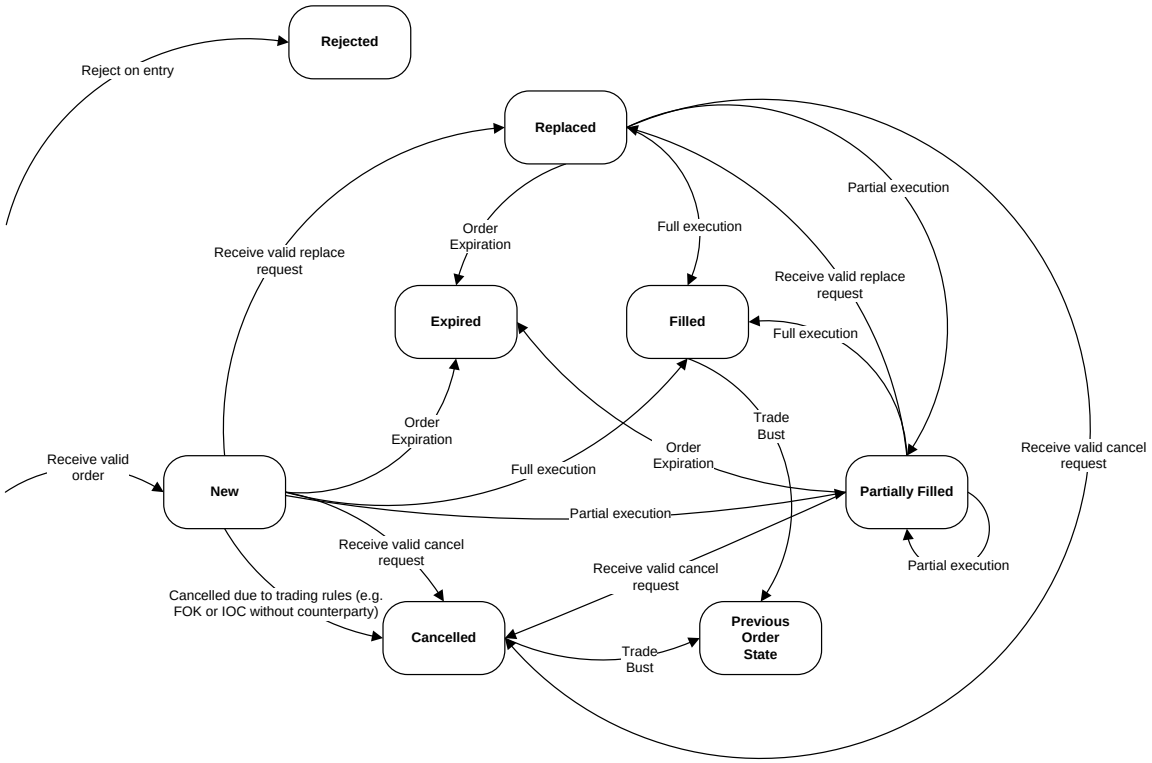

Figure 3 - Order State Transitions

6.5 Order Characteristics Modification/Removal

The various order characteristics behave differently with regard to modification. Some characteristics cannot be modified (e.g. order side), while others can be altered subject to market rules. The following table summarizes how and when various order characteristics can be changed:

Characteristic	Field	Tag	Add	Remove	Change
Order Type	OrdType	40			✓
Validity	TimeInForce	59			✓
Expire Date	ExpireDate	432	✓ (if modifying to GTD)	✓ (if modifying from GTD)	✓ (if TimeInForce is GTD)
Price	Price	44	✓ (if modifying to Limit or Stop Limit)	✓ (if modifying from Limit or Stop Limit)	✓ (if OrdType is Limit or Stop Limit)
Asset	Symbol	55			
Order Side	Side	54			
Exercise Block	PositionEffect	77	✓		
Stop Price	StopPx	99	✓ (if modifying to stop limit)	✓ (if modifying from stop limit)	✓ (if OrdType is Stop Limit)
Memo	Memo	5149	✓		✓
Minimum Quantity	MinQty	110	✓		
Order Quantity	OrderQty	38			✓
Disclosed Quantity	MaxFloor	111	✓	✓ (set value of tag MaxFloor to 0)	✓
Account	Account	1	✓	✓ (set value of tag AccountType to 38)	✓[3]
Entering Trader	PartyRole(36) + PartyID	452 + 458			
Executing Trader	PartyRole(12) + PartyID	452 + 458	✓		
Entering Firm	PartyRole(7) + PartyID	452 + 458			
Sender Location	PartyRole(54) + PartyID	452 + 458			
Clearing Firm	PartyRole(4) + PartyID	452 + 458	✓		✓
Investor ID	PartyRole(5) + PartyID	452 + 458	✓		✓

Only the fields that are being changed need to be sent in the replacement. Fields that are not sent are considered to be the same as the original order.

[3] Account Change/Removal might be unavailable to some participants due to specific market rules.

6.6 Impact of the changes on the order's priority

The changes in the orders may impact the priorities. The changes and the respective impacts are shown in the table below:

Characteristic	Impact on the priority
Price	Priority Loss
Stop price triggering	Priority Loss
Stop price limit	Priority Loss
Increase Quantity	Priority Loss
Decrease Quantity	Keep Priority
Minimum quantity	Priority Loss
Order Type	Keep Priority
Validity	Keep Priority
Account [4]	Keep Priority

6.7 Order Identification

EntryPoint supports a number of order identifiers which allows participants to keep track of order events during the whole order lifecycle.

6.7.1 Participant Issued Identifiers

Participant issued identifiers are assigned by market participants through the order lifecycle. They are discussed on the following subsections.

6.7.1.1 ClOrdID

The ClOrdID (11) field is the primary client side order identifier. It is initially assigned at order entry, and can be subsequently changed through the order lifecycle in modifications and cancelation requests. It must be unique among all active orders on a given instrument sent via a specific FIX session.

ClOrdID can be recycled after a given order reaches the end of its lifecycle (cancelled/completely filled). Although supported, this practice is not recommended, since it hinders troubleshooting.

6.7.1.2 OrigClOrdID

The OrigClOrdID (41) field is used in conjunction with the ClOrdID field and allows the client to implement client side order chaining, that is, to keep a history of client initiated order events. Each Modification/Cancellation request must have an associated OrigClOrdID. The following table depicts how chaining maintain order relation across subsequent operations.

Msg sent	Msg received	ClOrdID	Orig ClOrdID	OrderID	Price	Qty	Ord Status	Exec Type	Cum Qty	Leaves Qty
D		ABC1			12.00	1000				
	8	ABC1		ORD_1	12.00	1000	New	New	0	1000
G		MOD1	ABC1	ORD_1	12.00	1200				
	8	MOD1	ABC1	ORD_1	12.00	1200	Replaced	Replace	0	1200
G		MOD2	MOD1	ORD_1	12.00	1400				
	8	MOD2	MOD1	ORD_1	12.00	1400	Replaced	Replace	0	1400

[4] Account Change/Removal might be unavailable to some participants due to specific market rules.

6.7.1.3 ClOrdID/OrigClOrdID Chaining Rules

On outbound messages, EntryPoint echoes the ClOrdID/OrigClOrdID assigned in the inbound message:

Msg sent	Msg received	ClOrdID	OrigClOrdID	Price	Qty	ExecType
D		NEW1		12.00	1000	
	8	NEW1		12.00	1000	New
Rejected modification						
G		MOD1	NEW1	12.00	1200	
	9	MOD1	NEW1	12.00	1200	Rejected
Successful modification						
G		MOD2	NEW1	12.00	1400	
	8	MOD2	NEW1	12.00	1400	Replace

This behavior also allows participants to send the same ClOrdID/OrigClOrdID on all messages in a given order chain:

Msg sent	Msg received	ClOrdID	OrigClOrdID	Price	Qty	ExecType
D		ORD1		12.00	1000	
	8	ORD1		12.00	1000	New
Rejected modification						
G		ORD1	ORD1	12.00	1200	
	9	ORD1	ORD1	12.00	1200	Rejected
Successful modification						
G		ORD1	ORD1	12.00	1400	
	8	ORD1	ORD1	12.00	1400	Replace
Rejected cancelation						
F		ORD1	ORD1	12.00	1400	
	9	ORD1	ORD1	12.00	1200	Rejected
Successful cancelation						
F		ORD1	ORD1	12.00	1400	
	8	ORD1	ORD1	12.00	1400	Cancelled

6.7.1.4 Rapid Firing

Rapid firing consists in sending a stream of Modification requests without waiting from the acknowledgement of previous operations:

Msg sent	Msg received	ClOrdID	OrigClOrdID	Price	Qty	ExecType
D		NEW1		12.00	1000	
	8	NEW1		12.00	1000	New
Client sends two “rapid fire” modifications, both accepted						
G		MOD1	NEW1	12.00	1200	
G		MOD2	MOD1	12.00	1400	
	8	MOD1	NEW1	12.00	1400	Replace
	8	MOD2	MOD1	12.00	1400	Replace

Rapid firing is supported on all segments, provided that the session is not making use of the LiNe credit control system.

6.7.2 Exchange Issued Identifiers

Exchange issued identifiers are discussed on the following subsections.

6.7.2.1 OrderID

The OrderID (37) field is one of the exchange issued order identifiers. It is assigned by the matching engine on successful order entry and it remains the same during the entire order lifecycle. The identifier is guaranteed to be globally unique across all parameters. When multiple identifiers are available, OrderID has the highest precedence.

6.7.2.2 SecondaryOrderID

The SecondaryOrderID (198) field is an alternate identifier issued by the exchange systems. Assigned upon order acceptance, it can be used on subsequent modifications and cancellation operations. Uniqueness is guaranteed to be globally unique across all parameters. Differently from the OrderID, the SecondaryOrderID changes for every client initiated event, and when a disclosed quantity order is replenished. Changing allows SecondaryOrderID to properly hide disclosed orders through multiple replenishments. The following tables present the SecondaryOrderID behavior on various scenarios:

Example: Order is received, partially filled and completely filled (no change to SecondaryOrderID).

Msg sent	Msg received	ClOrdID	Symbol	OrderID	Secondary OrderID	OrdStatus	Exec Type	Comment
D		ABC1						New Order from trader.
	8	ABC1	WDLH08	1	1	New	New	Order is ack'ed by exchange.
	8	ABC1	WDLH08	1	1	Partially filled	Trade	Partial fill is received by trader.
	8	ABC1	WDLH08	1	1	Filled	Trade	Order is filled.

Example: Disclosed order is received, partially filled and completely filled (SecondaryOrderID changes upon replenishment).

Msg sent	Msg received	ClOrdID	Symbol	OrderID	Secondary OrderID	OrdStatus	ExecType	Comment
D		ABC1						New Order from trader.
	8	ABC1	WDLH08	1	1	New	New	Order is ack'ed by exchange.
	8	ABC1	WDLH08	1	1	Partially filled	Trade	Partial fill is received by trader.
Order is partially filled, and quantity is replenished.								
	8	ABC1	WDLH08	1	2	Partially filled	Trade	Partial fill is received by trader.
Order is partially filled, and quantity is replenished again.								
	8	ABC1	WDLH08	1	3	Filled	Trade	Order is filled.

Example: Order is received, and then modified (SecondaryOrderID changes upon modification).

Msg sent	Msg received	ClOrdID	Symbol	OrderID	Secondary OrderID	OrdStatus	ExecType	Comment
D		ABC1						New Order from trader.
	8	ABC1	WDLH08	1	1	New	New	Order is ack'ed by exchange.
Trader modifies order quantity.								
	8	ABC1	WDLH08	1	2	Replaced	Replace	Order was modified.
Order is filled.								
	8	ABC1	WDLH08	1	2	Filled	Trade	Order is filled.

6.7.3 Order Identifier Rules

The following rules apply to the order identifiers:

- ClOrdID/OrigClOrdID should be unique per FIX session and instrument. Uniqueness is checked among standing orders only.
- Participants can modify/cancel orders by OrderID, SecondaryOrderID, and OrigClOrdID. If more than one of them is present, OrderID has a priority over SecondaryOrderID, which has more priority than OrigClOrdID. FIX rules are observed, that is, ClOrdID/OrigClOrdID must be present on all modifications and cancellations requests.
- If OrderID or SecondaryOrderID are present on the Modification/Cancellation request, the engine uses them to look up the order. However, ClOrdID uniqueness rules must still be enforced. This means that a modification with a good OrderID but duplicated ClOrdID must be rejected. If the participant provides a valid OrderID, non-duplicate ClOrdID, and an invalid SecondaryOrderID/OrigClOrdID, the operation still completes since there is no harm to the lifecycle for subsequent operations. In this case, note that the engine will simply echo the values of SecondaryOrderID/OrigClOrdID as entered in the modification order.
- If an order is rejected upon entry, the rejection Execution Report contains a newly created OrderID.
- If a modification is rejected due to the order being unknown, the engine echoes whatever OrderID/SecondaryOrderID is present in the incoming message.
- In case OrderID/SecondaryOrderID is not provided in the incoming message, field OrderID/SecondaryOrderID is set to 'NONE'.

In the market data feeds, the OrderID (37) field actually contains the SecondaryOrderID (198) value for a given order.

7. Execution Report

The Execution Report message (tag 35=8) is used in the following scenarios:

- Confirm the receipt of an order;
- Confirm changes to an existing order (i.e. accept order cancel and order cancel replace requests);
- Relay order status information;
- Relay fill information on working orders (trades);
- Reject orders.

Each execution report contains two fields which are used to communicate both the current state of the order as understood by the broker and the purpose of the message:

- OrdStatus (39) used to convey the current status of an order and
- ExecType (150) used to identify the purpose of the Execution Report message.

7.1 Aggressor Indicator

The AggressorIndicator (1057) Boolean tag is returned in the Execution Report message to indicate whether the order initiator is an aggressor or not in the trade.

Possible values are:

- Y = Order initiator is aggressor
- N = Order initiator is passive

7.2 Average Execution Price

This functionality is not available through EntryPoint. Tag AvgPx (6) is kept for FIX compatibility only. Client systems should calculate the average execution price based on the execution prices sent in tag LastPx (31) of each fill's Execution Report message. EntryPoint will always send tag AvgPx (6) = 0.

7.3 Rejection Codes

BM&FBOVESPA will issue an Execution Report message (tag 35=8) with field ExecType set to "Rejected" (tag 150=8) for orders rejected by EntryPoint, the pre-trade risk control or the match engine. The code for the rejection is stated in tag OrdRejReason (103) and the actual text of the rejection is informed in tag Text (58).

In order to facilitate client system's handling of internationalization of error messages, BM&FBOVESPA makes a list of the possible reject codes available in the EntryPoint website, at:

http://www.bmfbovespa.com.br/en-us/services/download/EntryPointErrorCodes.pdf

7.4 Instrument Status

In order to provide the client systems the ability to identify the status of a given instrument when a trade takes place, without the need to cross information with market data feeds, the Execution Report message (tag 35=8) includes the following fields:

Tag	Tag name	Req'd	Data Type	Comment
336	TradingSessionID	N	String (1)	Identifier for Trading Session. Valid values: 1 - Regular Day Session phases. 6 - Used for both Before and After Hours Market phases (Non Regular Session)
625	TradingSessionSubID	N	String (3)	Identifier for the instrument group phase. Valid values: 2 - Pause 4 - Close 17 - Open 18 - Pre-Close 21 - Pre-Open 101 - Final Closing Call
6392	SecurityTradingStatusID	N	String (3)	Identifier for the instrument phase. Valid values: 2 - Trading Halt (Pause) 4 - No-Open (Close) 17 - Ready to trade (Open) 18 - Not available for trading (Forbidden) 20 - Unknown or invalid 21 - Pre-Open 101 - Final Closing Call 110 - Reserved

These fields are present only in Execution Reports for trades – identified by tag ExecType (150) = F (Trade – partial fill or fill). Including also Execution Report of Termo, UDS and trades inserted directly by the Market Operation desk.

So as to identify trades performed during the Opening and Closing auctions, for example, the content of field SecurityTradingStatus (6392) must be observed.

The following values of tag SecurityTradingStatus (6392) describe Opening and Closing auctions:

- 21 = Pre-Open
- 101 = Final Closing Call

In case field SecurityTradingStatus (6392) is not present in the Execution Report, it means the instrument follows the same phase of its group. Therefore, the value of tag TradingSessionSubID (625) must be considered instead.

The following values of tag TradingSessionSubID (625) describe Opening and Closing auctions, respectively:

- 21 = Pre-Open
- 101 = Final Closing Call

Tag TradingSessionID (336) is used to convey the phases of the market that can be reported as Regular or Non Regular session.

8. Participant Identification

In EntryPoint, participants are identified by the use of the Parties Component Block, a multipurpose identification construct. The component block contains the NoPartyIDs (453) repeating group. Each repetition contains one piece of relevant participant data. The relevant fields in the NoPartyIDs repeating group follows:

- PartyID (448): contains the actual participant identification
- PartyRole (452): qualifies the PartyID value. Thus, according to the PartyRole value, PartyID can represent a trader or firm, among other available roles.
- PartyIDSource (447): Identifies the issuer of the PartyID code. Currently, all participant codes are issued by BM&FBOVESPA, thus PartyIDSource is always set to 'D' (Proprietary/Custom Code).

The following table illustrates an example of a DMA provider (e.g. "XYZ") sending a New Order Single to BM&FBOVESPA, entered by trader "ABC":

<table>
<tr><th>Msg Sent</th><th>ClOrdId</th><th>Price</th><th>Qty</th><th>NoPartyIDs repeating group</th></tr>
<tr><td>D</td><td>ABC1</td><td>10.58</td><td>7000</td><td>NoPartyIDs = 3
<table>
<tr><th>PartyID</th><th>PartyIDSource</th><th>PartyRole</th></tr>
<tr><td>XYZ</td><td>D</td><td>54 (Sender Location)</td></tr>
<tr><td>ZZZ</td><td>D</td><td>7 (Entering Firm)</td></tr>
<tr><td>ABC</td><td>D</td><td>36 (Entering trader)</td></tr>
</table></td></tr>
</table>

The following table describes all available participant domains for the PartyRole field.

Party Role	Domain	Description	Applicable Segments		
			Equities	Derivatives	FX
Clearing Firm	4	Indicates the clearing member that will clear the trade.	N/A	Optional	N/A
Investor ID	5	Used for Self-Trading prevention at customer level. See section 14.5 for details.	Optional	Optional	Optional
Entering Firm	7	Broker identifier as assigned by BM&FBOVESPA.	Required	Required	Required
Executing Trader	12	Identifies the trader which is performing an action on behalf of another. Only used when trading on behalf. See section 8.1 for details.	Optional	Optional	N/A
Contra Firm	17	Broker identifier as assigned by BM&FBOVESPA used to indicate the counterparty brokerage firm in a Forward deal.	Required for forward deals, N/A for other securities	N/A	N/A
Entering Trader	36	Trader identifier. BM&FBOVESPA assigned for desk traders, free format for DMA.	Required	Required	Required

Sender Location	54	Identifies the order originator and DMA category. Desk traders always set this to 'BVMF', DMA connections must use the value assigned by BM&FBOVESPA.	Required	Required	Required
Session ID	55	BM&FBOVESPA may use this Party Role to convey the incoming FIX session's name in messages submitted to the Equities segment. **The use of this Party Role is reserved to the Exchange. Participants should not include it in order entry messages.**	N/A	N/A	N/A
Desk ID	76	Participants may use this field to identify the client associated with the given account number. This information may be used to correlate the order entry messages with the messages at the back-office and clearing systems.	Optional	Optional	Optional
Order Origination Session	1001	Participants may use this field to send an order cancelation on the Admin Session. BM&FBOVESPA may also assign this identifier in the Trade Cancel Reports.	N/A	N/A	N/A

DMA participants must avoid to provide Entering Trader (PartyRole (452) = 36) values that may coincide with trader IDs already used by the Mega Bolsa Station (range from 70 to 299).

8.1 Trading on Behalf

A desk trader in the role of 'desk supervisor' can trade on behalf of other desk traders and DMA clients. In order to do so, the supervisor must be registered in the exchange along with a visibility configuration which determines the set of participants that he represents. Provided that all previous conditions are met, the supervisor can send, modify, and cancel orders on behalf.

Whenever an on behalf operation is performed, the reply messages always go back to the order originator. Thus, if Supervisor S sent an order on behalf of trader T, the acknowledgment execution report is sent to S. T won't get a direct reply from the on behalf operation. However, if T has access to a drop copy session, T will receive the copy acknowledgment. Conversely, if the order was originally sent by T, all on behalf replies will flow back to T. S may get them via drop copy, if enabled.

Scenario 1: supervisor S sends an order on behalf of trader T

<table>
<tr><th>Msg Sent</th><th>Msg received</th><th>ClOrdID</th><th>Symbol</th><th>PartyIDs</th><th>Comment</th></tr>
<tr><td>D</td><td></td><td>ABC1</td><td></td><td>NoPartyIDs = 4
<table><tr><th>Party ID</th><th>PartyID Source</th><th>PartyRole</th></tr><tr><td>S</td><td>D</td><td>12 (Executing trader)</td></tr><tr><td>T</td><td>D</td><td>36 (Entering trader)</td></tr><tr><td>123</td><td>D</td><td>7 (Entering firm)</td></tr><tr><td>BVMF</td><td>D</td><td>54 (Sender Location)</td></tr></table></td><td>New Order from trader.</td></tr>
<tr><td></td><td>8</td><td>ABC1</td><td>WDLH08</td><td>Same as above</td><td>Order is ack'ed by exchange.</td></tr>
<tr><td colspan="6">Order control remain with S, and T cannot perform any operations upon the order</td></tr>
</table>

Scenario 2: supervisor S requests a security definition on behalf of trader T

Creating a user defined spread on behalf of a trader is also possible.

<table>
<tr><th>Msg sent</th><th>Msg received</th><th>SecurityReqID</th><th>PartyIDs</th><th>Comment</th></tr>
<tr><td>c</td><td></td><td>ABC1</td><td>NoPartyIDs = 4
<table><tr><th>Party ID</th><th>PartyID Source</th><th>PartyRole</th></tr><tr><td>S</td><td>D</td><td>12 (Executing trader)</td></tr><tr><td>T</td><td>D</td><td>36 (Entering trader)</td></tr><tr><td>123</td><td>D</td><td>7 (Entering firm)</td></tr><tr><td>BVMF</td><td>D</td><td>54 (Sender Location)</td></tr></table></td><td>Security Definition Request from trader.</td></tr>
<tr><td></td><td>d</td><td>ABC1</td><td>Same as above</td><td>Security Definition creation is confirmed by exchange.</td></tr>
</table>

Scenario 3: trader T sends an order, and supervisor S later modifies it

<table>
<tr><th>Msg sent</th><th>Msg received</th><th>ClOrdID</th><th>Symbol</th><th>PartyIDs</th><th>Comment</th></tr>
<tr><td>D</td><td></td><td>ABC1</td><td></td><td>NoPartyIDs = 3<table><tr><th>Party ID</th><th>PartyID Source</th><th>PartyRole</th></tr><tr><td>T</td><td>D</td><td>36 (Entering trader)</td></tr><tr><td>123</td><td>D</td><td>7 (Entering firm)</td></tr><tr><td>BVMF</td><td>D</td><td>54 (Sender Location)</td></tr></table></td><td>New Order from trader.</td></tr>
<tr><td></td><td>8</td><td>ABC1</td><td>WDLH08</td><td>Same as above</td><td>Order is ack’ed by exchange.</td></tr>
<tr><td>G</td><td></td><td>ABC2</td><td>WDLH08</td><td>NoPartyIDs = 4<table><tr><th>Party ID</th><th>PartyID Source</th><th>PartyRole</th></tr><tr><td>S</td><td>D</td><td>12 (Executing trader)</td></tr><tr><td>T</td><td>D</td><td>36 (Entering trader)</td></tr><tr><td>123</td><td>D</td><td>7 (Entering firm)</td></tr><tr><td>BVMF</td><td>D</td><td>54 (Sender Location)</td></tr></table></td><td>S modifies an order previously sent by T.</td></tr>
<tr><td></td><td>8</td><td>ABC2</td><td>WDLH08</td><td>Same as above</td><td>Execution report flows back to T. T remains in control of the order, and both S and T can modify or cancel the order on following operations.</td></tr>
</table>

9. Security Identification

In the order entry messages, the only required field to uniquely identify a security is Symbol (55). This tag conveys the human readable security identifier and it's available in the Security List message.

In order to trade securities listed in other venues (e.g. Globex); the SecurityExchange (207) tag should be used to specify the market to which the security belongs, using its proper MIC[5] code. If SecurityExchange it's not provided, BVMF is assumed as a default venue.

In order to maintain compatibility with of previous implementations of the order entry interface, tags SecurityID (48) and SecurityIDSource (22) are still accepted as part of the order entry inbound messages. However, those tags are not required and are not validated by the exchange systems. For the participants benefit, the tags will be echoed back in the outbound messages.

Participants that are starting a new EntryPoint development effort can omit SecurityID and SecurityIDSource from the order entry messages.

[5] Market Identifier Code, an International Standard (ISO 10383) used to uniquely identify Exchanges.

10. Access Categories

All four DMA categories are available in the equities and derivative segments. Each category has a new alpha-numeric code which indicates that a given connection belongs to a specific category and replaces the former Bovespa numeric ranges. The new alpha-numeric code must be sent in every message as a Sender Location Party Role.

The following table depicts all access categories, their associated codes and the correspondent Bovespa three digit numbers:

Access Category	Market Segment	Sender Location	Former Bovespa Numeric Range
Desk traders	Equities	BVMF	770-999
	Derivatives	BVMF	
Local	Equities	AUTO	N/A
	Derivatives	AUTO	
Give up Agent	Equities	REPS	310-359, 510-559
DMA1	Equities	DMA1	300-309, 360-399, 400-499, 500-509, 560-599
	Derivatives	DMA1	
DMA2	Equities	*Provider Code*	600-649
	Derivatives	*Provider Code*	
DMA3	Equities	DMA3	650-655
	Derivatives	DMA3	
DMA4	Equities	COLO0/COLO1	656-769
	Derivatives	COLO0/COLO1	

Although all categories share a large body of common behavior, the following differences apply:

- Desk traders and give up agents must specify a valid trader ID on all inbound operations. The trader code must be sent in a PartyID (448) field, along with an associated PartyRole (452) = 36 (Entering Trader). The trader ID is validated upon entry and, in case the value provided is not registered in the BVMF systems, the order will be rejected due to lack of permission.

- Only Desk Traders may modify the Account tag that was initially provided in the New Order Single.

- In the Derivatives and Equities segments, Desk Traders may omit the Account information.

DMA Providers (DMA2), sponsored by multiple Firms, must use different FIX sessions to submit orders from each Firm.

11. Memo

In order to provide a field that participants can use to submit a comment or a description about the current request, most of the messages in EntryPoint have been equipped with a customized tag called Memo (5149).

This tag is defined as a free format text field (limited in up to 50 characters) that may be used to convey client's relevant information.

The use of the Memo (5149) field is convenient because its content is always echoed in the reports. Additionally, as the information might have meaning only to its publisher, the content entered on this field is not visible to the counterparty.

Observe that the scope of tag Memo (5149) is restricted to the Order Entry scenario, which means that the information may be available around the Order Entry and Drop Copy gateways only. There is no guarantee that the text entered in tag Memo (5149) will reach other systems, such as in the clearing or post-trading areas. In this aspect, it is not recommended to use tag Memo (5149) as a key to correlate messages from the trading with data collected in the post-trading systems, for example.

If the participant needs to have the information reflected outside the scope of the trading environment, it is advised to consider using the PartyRole “76 - Desk ID”[6], that is also a free format text field, but which content can be used to add a description or comment to the client’s account number and therefore offered to external systems.

[6] Refer to the **Client Identification** section, in this document, for more information about the use of tag PartyRole = 76 (Desk ID).

12. Client Identification

12.1 Account Number

Client Identification should be sent in the Account (1) field. Exchange replies such as Execution Reports and Order Cancel Rejects will also echo the client identification in the Account (1) field.

The AccountType (581) tag acts as a qualifier of the account. For example, setting AccountType to '38' in a modification indicates that the account information should be removed from the order[7]. If AccountType is missing, the Account tag is interpreted as a regular account.

Tag	Tag name	Req'd	Data Type	Comment
1	Account	N	Int (12)	Account mnemonic.
581	AccountType	N	Int (2)	Type of Account associated with an order. Absence of this Tag causes Account to be interpreted as Regular Account. Valid values: 38 - Remove Account Information 39 - Regular Account

12.2 Account Annotation

There are scenarios where participants need to include an annotation in the order entry message exclusively to identify the client associated with a given account number. In most cases, this information is also used to correlate the order entry messages with the data in the back-office and clearing systems.

Although, in EntryPoint, the Account (1) field cannot be used to convey any type of text information (only numeric values are accepted), sum it has been implemented an efficient alternative to support this functionality.

In this context, users are advised to take advantage of PartyRole "76 - Desk ID" which provides a powerful and consistent method to allow participants to annotate the account. Observe the example below:

Msg	Account	ClOrdID	PartyIDs	Comment
D	1234	ABC1	NoPartyIDs = 4 Party ID / PartyID Source / PartyRole 123 / D / 7 (Entering firm) T / D / 36 (Entering trader) BVMF / D / 54 (Sender Location) JOHN / D / 76 (Desk ID)	Field Account is numeric only. PartyRole "Desk ID" is used as a label to the account number.

Due to compatibility with legacy systems, such as STM, the max length of field Desk ID is currently restricted to 8 characters. Should the max length be exceeded, the Desk ID information will be truncated and the rightmost ch aracters will be lost.

Differently from the information provided in the Memo (5149) field, which may also be used to enter specific client's information, the PartyRole (452) "76 - Desk ID" is guaranteed to circulate from the trading through post-trading environments.

[7] This functionality might be unavailable due to Market Rules (e.g. DMA participants cannot remove the account information). Please, refer to section Market Segment Specific Rules for market specific rules regarding account handling.

13. Market Segment Specific Rules

The following sections describe market specific rules.

13.1 BOVESPA Segment (Equities)

This section describes specific trading rules for the equities market.

13.1.1 Trading Hours

For a list of equities trading hours, sessions, and holidays, please visit: www.bmfbovespa.com.br and follow Rules ➔ Trading Hours ➔ Equities.

13.1.2 Client Identification

In the equities segment, all orders must be sent with an Account (1) field. Unspecified orders are rejected upon entry.

13.1.3 Orders triggering Instrument Freeze (frozen orders)

Orders that trigger an instrument freeze will be accepted by using the “suspended” state. The execution report indicating the suspension should be considered as an acknowledgement of order acceptance, i.e., it has the semantics of a ‘New’ execution report. When the instrument freeze is lifted, EntryPoint will send one or more fill notifications, or a cancellation execution report if the order is rejected as the instrument ‘thaws’. The following table illustrates the first scenario:

Order sent followed by freezing, partial fill and subsequent thawing of instrument

Message received (ClOrdID)	Message Sent (ClOrdID)	Exec Type	Ord Status	Order Qty	Cum Qty	Leaves Qty	Last Qty	Comment
New Order (X)				10,000				Order is sent by client system.
	Execution(X)	Suspend	Suspend	10,000	0	10,000		Order is accepted by the exchange, causing an instrument freeze.
	Execution(X)	Trade	Partially filled	10,000	6,000	4,000	6,000	Order is partially filled (quantity executed = 6,000)
	Execution(X)	Trade	Filled	10,000	10,000	0	4,000	Order is filled.

13.2 BM&F Segment (Derivatives)

This section describes specific trading rules for the derivatives market.

13.2.1 Message Flow

The following diagram depicts the overall message flow in the derivatives segment.

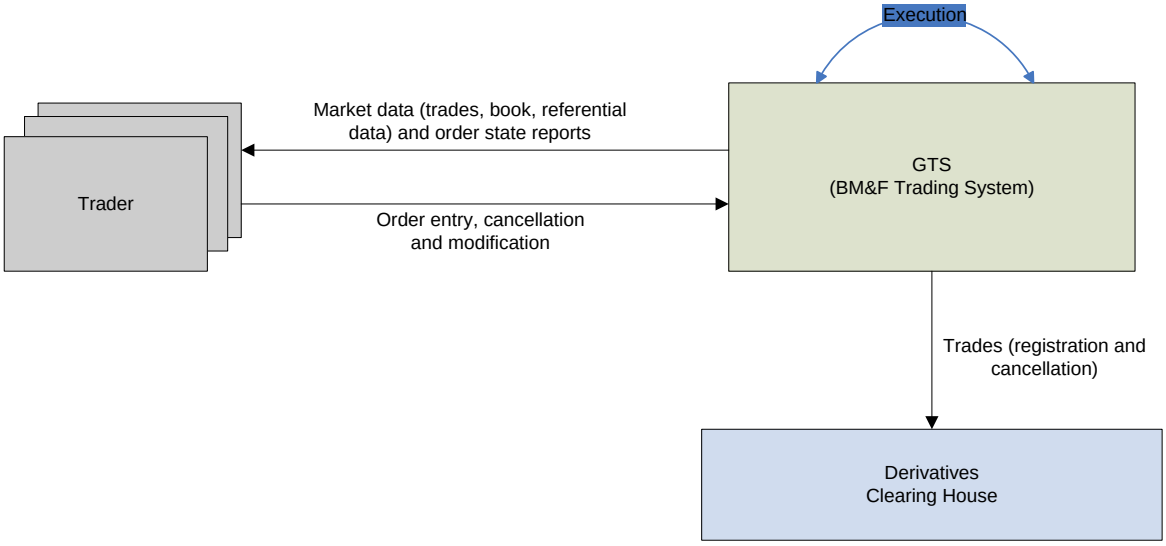

Figure 4 - Derivatives Clearinghouse message flow.

13.2.2 Trade Give-ups

A trade may be given up to another firm, i.e. the firm that carries the position is a firm other than the executing firm (the firm that puts the order in the market). Give-ups may be done post-trade (via the "BM&FBOVESPA Services" website) or pre-trade, indicating the give up in the order message.

The give-up indication is done in the order message by providing the account source number in the Account (1) field. Once the trade is made, the take up firm (the firm the trade was given up to) will receive notification and will accept or reject the trade. If the take up firm rejects the trade, the position is carried by the executing firm.

Example:

Customer "1234" was assigned, at the BM&FBOVESPA system, as the source account for entering firm = "GHI" which is linked to a give-up firm "DEF".

The target account number at firm "DEF" is "9898". Trader "ABC" is the trader that puts the order in the market.

Msg sent	ClOrdID	Price	Qty	Account	NoPartyIDs repeating group
D	ABC1	10.58	7000	1234	PartyID / PartyID Source / PartyRole ABC / D / 36 (Entering trader) GHI / D / 7 (Entering firm)

Since the target account number is not provided in the order message, the pre trade limit validation is performed on the source account "1234".

13.2.3 Closing a Short Options Position

All derivatives options are outright, that is, there is no implicit netting if the participant buys and sells the same amount of a given option. Should a participant wish to close (net) a short position, he may do so by buying the same amount of the same option, and set the PositionEffect (77) field to 'C' (Close).

Field PositionEffetct (77) must be used only with Derivative options. If this tag is assigned with other security types, the message may not be rejected. However, the participant's request might end up in an inconsistent state and might not be processed correctly.

13.2.4 Account Allocation Restrictions for DMA Customers

DMA originating orders should always be specified, that is, a valid account or give up link should be sent in the Account (1) tag.

Orders originated by DMA customers without account allocation or give-up code information will be rejected by EntryPoint. In addition, orders that are already available in the order book may not have their allocation account or give-up code modified. For changing these order characteristics pre-trade, the DMA customer will first need to cancel the order, and then issue a new order with the desired account.

13.3 Foreign Exchange (FX)

The following diagram depicts the overall message flow in the FX segment.

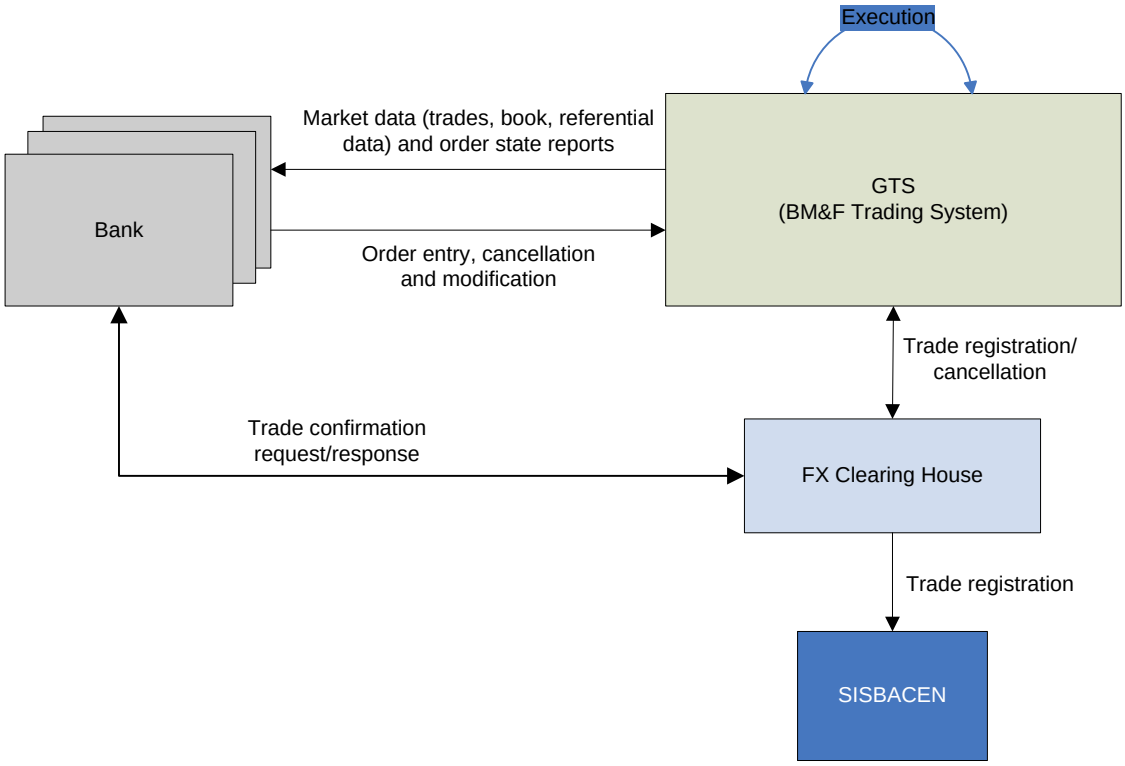

Figure 5 - FX Clearing House message flow.

13.3.1 Market Rules

The following applies to the FX segment:

- As determined by the current legislation, FX negotiation is restricted to banks. Thus, DMA is not available.
- All FX orders must contain an associated client. Post trade specification is not allowed.
- Self-trading is forbidden. If an incoming order can be matched, but the counterparty order contains the same client, the incoming order is rejected before matching takes place.
- Market data is provided as a blind screen, that is, the participating party is not specified in the message, i.e. the book information will not contain bank/broker information.
- Only Limit orders are supported (OrdType (40) = 2).

14. Advanced Functionalities

14.1 User-Defined Spreads (UDS)

User-Defined Spreads provide users of the electronic trading platform the ability to create strategies composed by their choice of leg instruments, leg ratio and leg side.

The main purposes of implementing this functionality are:

1. All or Nothing semantics, which guarantees that the participant will get fills on all strategy legs (respecting the aforementioned leg ratios) or none at all.
2. Allow customer to create customized spreads and have them immediately active (vs. calling operations staff to create the spread and have them available next day);
3. Reduce the number of pre-listed instruments since market participants have the possibility to create spreads on demand, based on actual needs, as opposed to the exchange trying to foresee what instruments are needed;
4. Improves the quality of the instrument listing, since spreads without activity are eliminated.

Currently, the creation of User-Defined Spreads is available only in the Equities segment. The estimative for provisioning this functionality in the Derivatives segment will be announced in an opportune date.

14.1.1 Creation Rules

There are some constraints that are enforced by the system and must be observed during the creation of a UDS, such as:

- Instruments used to create a UDS must reside on the same engine.
- The first listed instrument will be used to determine the engine. In addition, instruments listed in the Black List cannot be used as strategy legs.
- The system does not allow user-defined strategies that use the same instrument as different legs. For example:
  - UDS-1: Buy ACME4; Sell ACME4
  - UDS-2: Buy ACME4; Sell 2 ACME3; Buy ACME4
- The system also does not allow a strategy to be created as the opposite of an existing one. For example:
  - UDS-1: Buy ACME4; Sell ACME3
  - UDS-2: Sell ACME4; Buy ACME3

  In this case, instead of creating UDS-2, the brokerage firm should sell UDS-1 to achieve the same effect.
- All strategies are created from a buyer's perspective. In the previous example, selling UDS-1 means to sell ACME4 and to buy ACME3, thus doing the opposite of the strategy creation.
- The system does not allow strategies with underlying legs of different Tick Sizes.

14.1.2 Expiration Date

A UDS is valid for at least seven days. If no activity occurs during this timeframe, the instrument will be deleted.

14.1.3 Security Strategy Types

There are a number of strategies that may be structured using the UDS functionality, such as Box, Straddle and Butterfly. Please refer to **Appendix D:** for examples and a detailed list of strategy types supported by the trading platform.

14.2 Exercise & Blocking

The Options Exercise & Blocking functionality allows market participants to manage their positions for Share Options and Index Options on the PUMA Trading System.

Option position is the balance resulting from one or more operations with options from the same series, executed on behalf of the same investor and through the same brokerage firm. Depending on the nature of the balance, the position may be writing or holding and depending on the collateral provided, covered or naked.

The estimative for provisioning the Exercise & Blocking functionality to the Derivatives segment in the EntryPoint order entry interface will be announced in an opportune date.

14.2.1 Exercise

Exercise may be defined as whenever holders of options contracts put into effect their right to buy or sell the option's underlying asset at the strike price.

During the trading session, BM&FBOVESPA receives exercise requests, as a Position Maintenance Request (35=AL), and validates business rules prior the execution of the request. If the request is valid, BM&FBOVESPA creates trades on the exercise instrument and updates the current client's position.

If at any time an exercised trade is busted by market operations, the Issuer and the Holder positions are reallocated at the system and the Issuer will be again eligible to be exercised.

14.2.1.1 Exercise Instrument

Every option (except Index) traded at BM&FBOVESPA has one correspondent non-tradable "exercise instrument".

For American Options, the exercise instrument is created in D+1 from the acquisition date. For European Options, the exercise instrument is created only at maturity date to prevent exercising prior to maturity.

A holder position is only eligible to exercise at D+1 from the acquisition date. Any exercise request from a position acquired at the same day will be immediately rejected by the Exchange.

14.2.1.2 Threshold Amount

The value of tag ThresholdAmount (834), in the PositionMaintenanceRequest (35=AL), may be used to indicate the minimum earnings an options contract holder expects to profit by exercising his position.

14.2.2 **Automatic Blocking**

Automatic Blocking is also known as Blocking by Purchase. It occurs when clients prevent the assignment of a portion or the entirety of their positions, upon prior purchase of an equity option from the same series previously issued.

The position can be partially or totally blocked, depending on the quantity issued and the quantity bought. To consider a position blocked, the following conditions must be considered:

- Account number must be the same
- Option series must be the same
- Brokerage Firm must be the same
- Quantity does not need to be the same, but the quantities bought and already exercised will determine if the position is partially or totally blocked

All automatic blocking will be notified with a Position Maintenance Report (35=AM) being sent to the Participant right after the purchase.

14.2.3 **Blocking Specification**

The Blocking Specification functionality has been developed to provide the ability to the system, whenever necessary, to prevent the assignment of an Options contract position.

It allows the customer to send an Allocation Instruction (35=J) message associated to a particular trade in order to mimic an Automatic Blocking.

14.3 Scheduled Exercise on Stock and Equity EFT's Options

Additionally to the Exercise functionality presented above, where participants must explicitly request the exercise of an option contract, BM&FBOVESPA also offers a feature that allows participants to schedule the series of options intended to be exercised.

The Scheduled Exercise on Stock and Equity EFT's Options functionality takes in consideration a threshold percentage that informs how much *in the money* the series must be in order to be exercised.

The Exercise will be triggered only if the specified *in the money* percentage is equal or better than the underlying asset price, which will be calculated by BM&FBOVESPA as a weighted average in a given time interval.

Participants will be able to register the Scheduled Exercise only at the maturity date of the options contract.

14.3.1 FIX Tags Usage

14.3.1.1 FIX Tag PosTransType (709)

In order to indicate a Scheduled Exercise, participants must set tag PosTransType (709) in the PositionMaintenanceRequest (35=AL) message with the following value.

Tag	Tag name	Value	Comment
709	PosTransType	102 - Scheduled Exercise	Identifies the type of position transaction.

14.3.1.2 FIX Tag ThresholdPercent (35048)

Scheduled Exercise requests must provide an *in the money* percentage value that will be used to decide whether the option must be exercised.

This percentage value must be provided in tag ThresholdPercent (35048).

Tag	Tag name	Value	Comment
35048	ThresholdPercent	Must be greater than or equal to zero.	A percentage value that informs how much *in the money* the series must be in order to be exercised.

The value provided in tag ThresholdPercent (35048) must be a decimal number between 0 and 1, otherwise the request will be rejected.

The use of tag ThresholdPercent (35048) is restricted to Scheduled Exercise requests. In case tag ThresholdPercent (35048).

14.3.2 **Examples**

In this section, we present some examples of the Scheduled Exercise on Stock and Equity EFT's Options functionality.

14.3.2.1 Scheduled Exercise – All Requests Successful

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for Scheduled Exercises to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA preset time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1.000 ACMEL20 Options for client 999-9
- Exercise request for 1.000 ACMEL20 Options for client 999-8
- Exercise request for 1.000 ACMEL20 Options for client 999-7
- *In the money* percentage for all requests is 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h50 PM:

- All requests from broker X get executed.

14.3.2.2 Scheduled Exercise– Part of Requests Successful

At option's maturity date:

- Exercise group status is open.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45→ 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Option characteristics: ACMEL91; Call; Strike Price R$ 20.10
- Option characteristics: ACMEL92; Call; Strike Price R$ 20.20

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL91 Options for client 999-8
- Exercise request for 1000 ACMEL92 Options for client 999-7
- In the money percentage for all requests equals 1,00%
- For ACMEL20, event will trigger if underlying weighted average price is R$ 20.20 or higher.
- For ACMEL91, event will trigger if underlying weighted average price is R$ 20.30 (R$ 20.3010 truncated to two decimal numbers) or higher.
- For ACMEL92, event will trigger if underlying weighted average price is R$ 20.40 (R$ 20.4020 truncated to two decimal numbers) or higher.

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h55 PM:

The following exercise requests are executed:

- Exercise request for 1000 ACMEL20 Options for client 999-9

The following exercise requests are triggered but expire:

- Exercise request for 1000 ACMEL91 Options for client 999-8
- Exercise request for 1000 ACMEL92 Options for client 999-7

Though all options have in the money strike prices, two of the requests expires since the specified in the money percentage (1.00%) is not achieved.

14.3.2.3 Requests with Several *In the Money* Percentages and Series

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Option characteristics: ACMEL91; Call; Strike Price R$ 20.10
- Option characteristics: ACMEL92; Call; Strike Price R$ 20.20

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1.000 ACMEL20 Options for client 999-9
  - In the money percentage equals 1,00%
  - Event will trigger if underlying weighted average price is R$ 20.20 or higher.

- Exercise request for 1.000 ACMEL91 Options for client 999-8
  - In the money percentage equals 0,50%
  - Event will trigger if underlying weighted average price is R$ 20.20 (R$ 20.2005 truncated to two decimal numbers) or higher.

- Exercise request for 1.000 ACMEL92 Options for client 999-7
  - In the money percentage equals 0,25%
  - Event will trigger if underlying weighted average price is R$ 20.25 (R$ 20.2505 truncated to two decimal numbers) or higher.

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h55 PM:

The following exercise requests are executed:

- Exercise request for 1.000 ACMEL20 Options for client 999-9
- Exercise request for 1.000 ACMEL91 Options for client 999-8

The following exercise requests are triggered but expire:

- Exercise request for 1.000 ACMEL92 Options for client 999-7

14.3.2.4 *In the Money* Percentage Not Met

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.,20 or higher

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
2000	R$ 20,00	12:45
5000	R$ 20,10	12:46
200	R$ 20,15	12:47
100	R$ 20,20	12:48
1500	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.10 (R$ 20.1051 truncated to two decimal numbers).

At 12h55 PM:

Once the underlying weighted average price is lower than the specified *in the money* percentage (1.00%) all exercise requests expire.

14.3.2.5 Manual Exercise after Request Period

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20,00

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher.

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
2000	R$ 20,00	12:45
5000	R$ 20,10	12:46
200	R$ 20,15	12:47
100	R$ 20,20	12:48
1500	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.10 (R$ 20.1051 truncated to two decimal numbers).

At 12h55 PM:

Exercise requests are sent successful and Broker X receives an Execution Report indicating that its exercises expired once the underlying weighted average price appraised is R$ 20.10 and, by all means, lower than the Broker in the money percentage prior condition (1.00%).

However, between 12h55 and 12h59, the following trades in ACME4 occur:

Quantity	Price	Hour
3000	R$ 20,15	12:55
500	R$ 20,15	12:56
1000	R$ 20,20	12:57
2000	R$ 20,25	12:58
8000	R$ 20,20	12:59

At 12h59 PM:

Broker X decides to exercise, manually:

- Exercise 1000 ACMEL20 Options for client 999-9
- Exercise 1000 ACMEL20 Options for client 999-8
- Exercise 1000 ACMEL20 Options for client 999-7

All clients positions are successfully executed through manual intervention.

14.3.2.6 Scheduled Exercise of Partial Quantity

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Client's total position: 1.000 options each.

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher

At 11h30 AM:

Broker X performs the following exercises, manually:

- Exercise 500 ACMEL20 Options for client 999-9
- Exercise 500 ACMEL20 Options for client 999-8
- Exercise 500 ACMEL20 Options for client 999-7
- Scheduled Exercise requests remain disclosed and in force

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h55 PM:

The following Scheduled exercise requests from Broker X are treated:

- Exercise request for 1.000 ACMEL20 Options for client 999-9
- Exercise request for 1.000 ACMEL20 Options for client 999-8
- Exercise request for 1.000 ACMEL20 Options for client 999-7

And the following executions for Broker X takes place:

- Exercise 500 ACMEL20 Options for client 999-9
- Exercise 500 ACMEL20 Options for client 999-8
- Exercise 500 ACMEL20 Options for client 999-7

Broker X gets an Execution Report for each exercise informing the Scheduled exercise quantity was in fact 500 and not the original 1000.

14.3.2.7 Scheduled Exercise Request Expires due to Manual Exercise

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Client's total position: 1.000 options each.

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher

At 11h30 AM:

Broker X performs the following exercises, manually:

- Exercise 1000 ACMEL20 Options for client 999-9

The manual exercise occurs and the Scheduled exercise request for 1.000 ACMEL20 options expires immediately since it represents the client's total position.

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h55 PM:

The following Broker X Scheduled exercises befall:

- Exercise 1000 ACMEL20 Options for client 999-8
- Exercise 1000 ACMEL20 Options for client 999-7

14.3.2.8 Scheduled Exercise Request Expires due to Multiple Manual Exercises

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Client's total position: 1.000 options each.

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 1000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher

At 11h30 AM:

Broker X exercises manually:

- 100 ACMEL20 options for client 999-9.
- Scheduled Exercise request for client 999-9 rests disclosed and in force.

At 11h45 AM:

Broker X exercises manually:

- 100 ACMEL20 options for client 999-9.
- Scheduled Exercise request for client 999-9 rests disclosed and in force.

At 12h00 PM:

Broker X performs the following exercise, manually:

- 800 ACMEL20 options for client 999-9.

At this point the Scheduled exercise request for client 999-9 expires instantly, since the amount exercised adds up to 1000 which represents the client's total position.

From 12h45 PM to 12h50 PM:

- During time interval defined for the weighted average price calculation, the following trades on ACME4 take place:

Quantity	Price	Hour
1000	R$ 20,00	12:45
2000	R$ 20,10	12:46
500	R$ 20,15	12:47
500	R$ 20,20	12:48
9000	R$ 20,25	12:49

- Calculated underlying weighted average price is R$ 20.20 (R$ 20.2019 truncated to two decimal numbers).

At 12h55 PM:

The following Broker X Scheduled exercises transpire:

- Exercise 1.000 ACMEL20 Options for client 999-8
- Exercise 1.000 ACMEL20 Options for client 999-7

14.3.2.9 Scheduled Exercise Request Rejected

At option's maturity date:

- Exercise group status is *open*.
- Underlying group status is *open*.
- BM&FBOVESPA time interval for baskets to be sent: 10h00 → 12h45
- BM&FBOVESPA time interval for underlying weight average price calculation: 12h45 → 12h50.
- BM&FBOVESPA time interval for Scheduled Exercise event: 12h55.
- Exercise time limit: 13h00.
- Option characteristics: ACMEL20; Call; Strike Price R$ 20.00
- Client's total position: 1000 options each.

At 10h30 AM:

Broker X sends several requests of Scheduled Exercises with the following features:

- Exercise request for 2000 ACMEL20 Options for client 999-9
- Exercise request for 1000 ACMEL20 Options for client 999-8
- Exercise request for 1000 ACMEL20 Options for client 999-7
- In the money percentage for all requests equals 1.00%
- Event will trigger if underlying weighted average price is R$ 20.20 or higher

Scheduled Exercise request for client 999-9 (2000 ACMEL20 options) gets rejected since the informed quantity is higher than existing for exercise.

All remaining requests are properly disclosed and in force.

14.4 Forward Declaration/Acceptance ("Termo")

The Forward (also known as "Termo") Declaration/Acceptance model allows participants to record an out of band, pre-arranged deal, in the exchange environment.

Forward contracts are also available in the Derivatives segment. However, they are traded in the open order book model. The declaration/acceptance model is currently restricted to the Equities segment.

The Quote Request FIX message is used within the context of this Forward transaction in which two parties have completed a deal outside the Exchange and are initiating the negotiation process to formalize and execute this operation on the Exchange.

This is done privately between these two counterparties so the Quote Request submitted by the Initiator will be directed to the Respondent.

DMA participants can only initiate the negotiation but cannot be the counterparty to Forward contacts. Desk Traders can either declare or accept Forward deals.

14.4.1 Forward Types

There are four types of forward contracts that can be entered. The following table describes each one of these contracts:

Type	Description
Common forward	Forward trade to be physically and financially settled at the agreed face value.
Flexible forward	Forward trade that has as a specific feature that differentiates it from common forward; the possibility of enabling the forward purchaser to replace underlying stock of the initially agreed contract.
Dollar forward	Forward trade whose contractual price will be corrected daily by the variation of the average foreign exchange rate of the Brazilian Real (BRL) against the US dollar (USD), as of the trade day to the closing day, excluding first and last.
Index points forward	Forward trade that allows the secondary trading of forward contracts, in which the financial settlement amount is calculated by converting the value of the index points into local currency.

14.4.2 Forward + Cash (“Termo Vista”)

An alternative modality of forward contract is the Forward + Cash, also known as “Termo Vista”, which is a type of transaction that involves an operation in the forward market with its inclusion on the cash market, inverting the buyer and seller, i.e., the forward buyer becomes the cash seller, and the forward seller becomes the cash buyer.

14.4.3 Forward + Registered Cash (“Termo Vista Registered”)

Similarly, Forward + Registered Cash also involves an operation in the cash market. The difference is that, in the Forward + Registered Cash modality, the customer indicates the id of a previous cash trade in the declaration message.

At the end of the forward negotiation, a cash trade is executed inverting the buyer and seller, i.e., the forward buyer becomes the cash seller, and the forward seller becomes the cash buyer. Exactly how it happens in the Forward + Cash modality.

14.4.4 Security Code

Every security allowed to be traded at the exchange forward market has one correspondent non-tradable symbol: e.g. ACME4 has a non-tradable instrument ACME4T (“T” means Termo).

It is not possible to buy or sell it, except by sending and receiving a declaration. The security codes used in the forward market are:

Type of forward	Letter	Example
Common	T	ACME4T
Flexible	S	ACME4S
Dollar forward	D	ACME4D
Index points	T	ACME51T

14.4.5 Instrument States

Forward instruments are in Forward-specific groups. The group schedule has the following states:

- Closed
- Open (scheduled to open simultaneously with the underlying)
- Forbidden
- A declaration can only be accepted if the Forward instrument is in the Open state.

14.4.6 Quote Lifecycle

Every action taken in a Forward deal such as Declaration, Acceptance, Cancellation, Refusal or Expiration is confirmed by a Quote Status Report (35=AI) message.

The outcome of the operation is expressed in the QuoteStatusReportType (35005) field which can be New, Accept, Reject or Expired. The actual status of the Quote is conveyed by the QuoteStatus (297) field and can assume values such as Pending, Accepted, Cancelled, Quote Not Found, Pass or Expired. Please refer to **Appendix C:** for a detailed list of QuoteStatusReportType and QuoteStatus transitions.

The following diagram depicts the Quote state transition according to the agent that triggered the action and its respective outcome:

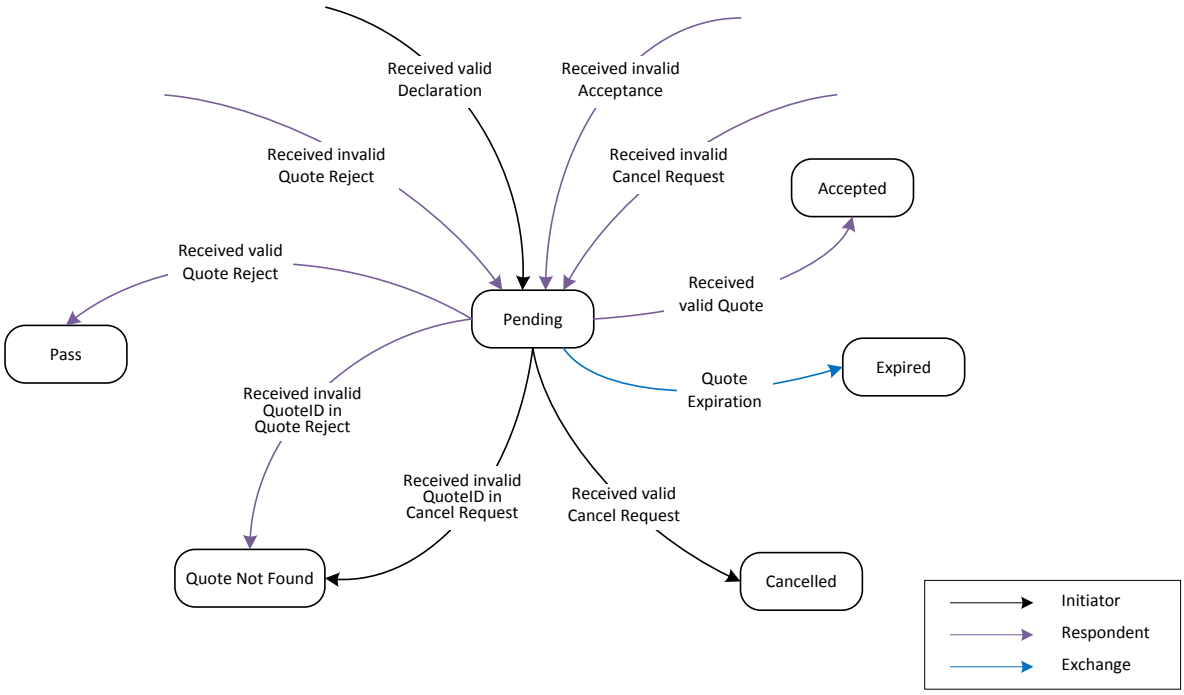

Figure 6 - Quote Status Transitions

14.4.6.1 Declaration

In the Declaration, the Forward deal is entered by the Initiator and received by the counterparty. It contains all the information (e.g. interest rate, price, quantity) that needs to be analyzed by the Respondent. Either buyer or seller can enter the declaration.

14.4.6.2 Cancelation

Before the Forward being accepted by the Respondent, the Initiator can send a Cancelation and terminate the deal.

14.4.6.3 Acceptance

In the Acceptance, the Respondent can either accept or reject the Forward deal. After the respondent accepts the Forward deal, BM&FBOVESPA generates trade execution reports for the Initiator and Respondent on their respective FIX sessions. In order to fulfill the deal, the system might generate more than one trade execution report per counterparty.

14.4.6.4 Refusal

If the Respondent doesn't agree with the terms presented in the Declaration, the Forward can be refused and the Initiator will be notified. The Initiator will need to send a new Declaration in order to reintroduce the deal.

14.4.6.5 Expiration

By the end of the day, all Declarations that have not been accepted are expired and the Forward deal is automatically terminated by the Exchange.

14.4.7 Contract Details

The following table describes some of the fields used in the FIX messages. These fields represent important concepts in a Forward deal and their meaning are presented below:

Field	Description
PrivateQuote (1171)	When trading a Forward contract, this field must be set as "Y" to specify that this quote is private, i.e. available to a specified counterparty only.
SettlType (63)	Determines who has the power to anticipate the settlement on the deal. Can be Buyer (Regular), Seller or Mutual. Mutual means agreed between the parties.
DaysToSettlement (5497)	Deadline for completing the Forward deal.
FixedRate (5706)	Interest to be paid by the forward buyer and received by the forward seller, in proportion to the agreed days to settlement.
ExecuteUnderlyingTrade (35004)	Specifies if a simultaneous trade of the underlying security is to be performed. Used to indicate Termo Vista.

Due to system's limitations, it's advised that a Firm should not send two identical Forward contracts to the same Counter Firm during the same day.

14.5 Self-Trading Prevention

Self-Trading Prevention at customer level is a functionality that aims to restrict matching between buying and selling orders from the same customer, regardless of Broker/Firm.

For this purpose, the customer must be identified with a unique Investor ID, included within the order message. The use of this unique identifier is optional and it is up to the customer to provide this information within the Parties block in the FIX message.

Note that Investor ID is not the same as the customer's account number nor is there necessarily a one-to-one relationship mapping between Account and Investor ID values.

14.5.1 Party Identification

As presented below, PartyID (448) field must be used to convey the unique customer identifier used by Self-Trading Prevention and PartyRole (452) must be assigned with value 5 – "Investor ID".

Tag		Tag Name	Data Type	Value
453		NoPartyIDs	NumInGroup	
→	448	PartyID	String	Investors must provide here their unique customer identifier in order to enable Self-Trading prevention.
→	447	PartyIDSource	Char	D
→	452	PartyRole	Int	5

It's important to note that Self-Trading Prevention at customer level works only if both aggressor (incoming) and aggressed (resting) orders are assigned with the same unique Investor ID.

However, two orders – on opposite market sides – with the same unique customer identifier can rest in the book simultaneously, as long as there is no potential match between them.

Self-Trading Prevention rules at the customer level do not apply for auction, match events that trigger an auction, orders entered on-behalf by GSN and UDS legs.

14.5.2 **Investor ID**

In order to guarantee the oneness on Investor IDs, BM&FBOVESPA adopted the following convention to define the customer identifier for each participant:

- Brazilian residents and corporate investors must use the 8 leftmost digits of their CNPJ;
- A Brazilian resident and individual investor uses the whole CPF as a unique customer identifier;
- Non-resident investors must use a six-digit code extracted from their individual investor ID. The whole code is formatted as AAAAA.BBBBBB.CCCCCC.X-Y, where only the CCCCCC part is used.

The table below presents some examples of Investor IDs:

Investor	Document Type	Document Number	Unique Customer Identifier
Corporate resident	CNPJ	01.234.567/0001-23	01234567
Individual resident	CPF	012.345.678-90	01234567890
Non-resident investor	Investor ID	01234.567890.123456.7-8	123456

Note: All Investor ID values above should be provided in tag PartyID (448) as numeric only (i.e. exclude alphanumeric characters such as dashes and dots).

Self-Trading Prevention is available through the EntryPoint API in all segments.

14.6 Message Retransmission

14.6.1 Resend Request

When using the ResendRequest (35=2) functionality to request the retransmission of a set of application messages, the maximum number of messages allowed in the request will be limited by the system. This limitation is configurable and intended to prevent performance degradation in the system due to intensive use of the Resend Request feature.

The limit will be set to 10,000 messages per request. Should the client need the retransmission of a larger number of application messages, it is highly recommended to use the Message Replay service.

If the range of the Resend Request (35=2) exceeds the stipulated limit, the system will accept the request, however, only the limited number of messages will be retransmitted.

In this case, the SequenceReset (35=4) message sent at the end of the retransmission will contain a custom tag PossMissingApplMsg (35033) = Y to indicate that some application messages are possibly missing.

Tag PossMissingApplMsg (35033) conveys important information regarding whether the Resend Request was completely satisfied or the client might connect to the Message Replay gateway in order to recover the complete set of messages.

Tag	Tag name	Req'd	Data Type	Comment
35033	PossMissingApplMsg	N	Boolean	Returned when the range of messages informed in a Resend Request is greater than the maximum range permitted. Possible values: Y = Indicates that the range of messages retransmitted after a Resend Request may not include all the application messages contained in the original range requested. N = No application messages are missing.

In order to simplify implementation, clients should consider always request the retransmission of application messages to the Message Replay service, instead of using the Resend Request functionality, even if the number of messages requested is lesser than the stipulated limit.

14.6.2 **Message Replay**

BM&FBOVESPA has implemented a new service dedicated mainly to support the requests for the retransmission of large amounts of messages. This feature will be particularly appreciated by late joiner Firms who need to get up to date with all operations performed by their clients during the current trading session.

The FIX component block ApplicationSequenceControl will be assigned to all application messages being resent to the client through the Message Replay service. The reason is to preserve the original values contained in the messages prior to the retransmission.

Tag ApplID (1180) conveys the original TargetCompID (56) and tag ApplSeqNum (1181) the original MsgSeqNum (34).

Tag	Tag name	Req'd	Data Type	Comment
1180	ApplID	Y	String (50)	Original TargetCompID of message. Identifies the session with which a message is associated. Assigned by Message Replay plug-in when resending application message.
1181	ApplSeqNum	Y	String (9)	Original MsgSeqNum of message. Assigned by Message Replay plug-in when resending application message.

Message Replay consists of a separate gateway to which participants must establish a FIX session connection. The retransmission is requested using a set of FIX 5.0 messages which have been adapted to the FIX4.4 specification in order to be used in the standard FIX session layer.

In the request is necessary to inform the rage of the messages to be replayed, based on the value of tag MsgSeqNum (34), and also the identification of the FIX session from which the messages were originally transmitted.

Find below the main aspects of the messages systems will need to handle in order to use this functionality.

14.6.2.1 ApplicationMessageRequest (35=BW)

Customers may send this message to request the retransmission of messages previously sent on a given FIX session indicated by tag RefApplID (1355).

The messages to be replayed are based on the tag MsgSeqNum (34) range specified in tags ApplBegSeqNum (1182) and ApplEndSeqNum (1183).

Tag		Tag name	Req'd	Data Type	Comment
1351		NoApplIDs	Y	NumInGroup (1)	Specifies number of application id occurrences. It must be always 1.
→	1355	RefApplID	Y	String (50)	Reference to the FIX session identifier.
→	1182	ApplBegSeqNum	Y	Seqnum (9)	Beginning range of application sequence numbers.
→	1183	ApplEndSeqNum	Y	Seqnum (9)	Ending range of application sequence numbers. Zero means sequence number of last message in transmission.

14.6.2.2 ApplicationMessageRequestAck (35=BX)

This message is sent by BM&FBOVESPA to acknowledge an ApplicationMessageRequest (35=BW). It provides only a status on the request (i.e. whether successful or not), this message does not provide the actual content of any message to be resent.

Tag	Tag name	Req'd	Data Type	Comment
35021	ApplResponseStatus	Y	Int (1)	Used to indicate the status for each Application Message Request. Valid values: 0 - Request Successfully Processed 1 - User Not Authorized For Application 2 - Invalid Range Requested 3 - Prior Application Request In Progress 4 - Application Temporarily Unavailable

14.6.2.3 ApplicationMessageReport (35=BY)

This message is used for three different purposes:

- to indicate a gap fill;
- to indicate that the last message has been sent;
- to indicate that an error occurred during message retransmission.

The purpose of the Application Message Report is indicated in tag ApplReportType (1426).

Tag	Tag name	Req'd	Data Type	Comment
1426	ApplReportType	Y	Int (1)	Type of report. Valid values: 0 - Reset ApplSeqNum to new value specified in tag ApplNewSeqNum (1399) 3 - Application Message resend completed 4 - Application Message resend error

A Sequence Reset gap fill is sent when there's a set of administrative messages within the range requested. The system does not replay administrative messages, but an Application Message Report is sent with tag ApplNewSeqNum (1399) indicating the expected sequence number for the next application message.

14.7 Market Protections

Market Protections are parameters selected by the participants to help them to reduce their risk. These parameters are set for each FIX session.

When participants choose to use this feature, the PUMA Trading System begins to monitor the protected trading groups and when a given limit is reached (or exceeded in certain scenarios) the trading platform triggers the Protected Mode, rejecting new messages and canceling the remaining orders for all instruments associated with the protected group.

It's important to observe that although this functionality is intended to mitigate potential losses, establishing a protection threshold, some conditions may prevent the correct operation of the feature and there is no guarantee that all resting orders will be successfully cancelled.

For example, in case the market or the instrument is in a state that does not allow order cancelations, such as *Pre Close* and *Close* states, orders for that instrument will not be cancelled.

Moreover, the Market Protections cannot be activated mid-execution, thus any protection is subject to having its value exceeded. In section 15.9.5 we present an example where a protection threshold is exceeded.

Market Protections are available on all asset classes (cash, stock options, index options, strategies, and derivatives). Please contact BM&FBOVESPA Trading Support Department (SSN) for more details on this feature.

14.7.1 Protection Types

There are a number of protection types available for market makers to use. The table below summarizes each one:

Protection	Description
New Order Fill Protection	Number of new order fills within a group during the time interval.
Execution Protection	Number of actual fills, including partial fills, within a group during the interval.
Traded Quantity Protection	Gross quantity of all instruments traded within a group during the interval.
Buy/Sell Protection	Net count of buys (+1) and sells (-1) traded within a group during the interval.
Delta Protection	Aggregate (combined + and -) delta values from each execution (Fill) are validated against the specified delta protection value.

As indicated in the table above, a Time Interval is in place for each protection type. The Time Interval only starts after a trade takes place, it does not continuously run throughout the session. If the elapsed time from the Time Interval start is greater than the assigned Time Interval Value, the counters of the enabled protections are automatically reset to zero.

Following, we describe the Market Protections in details and present examples for each different type.

14.7.1.1 New Order Fill Protection

The customer selects a threshold for the number of new order fills per Session, within a Time Interval.

The Protection calculation is based on the whole order and not the number of executions the order may generate. An entire order that is fully filled will be shown as one in the New Order Fill Protection count. Buys and Sells both increases the New Order Fill counter by '1'.

Example

Consider the situation described below:

- A customer sets their New Order Fill protection limit = 5 in Group 10
- BVMF sets the Time Interval = 15 seconds
- A customer has multiple resting orders for Group 10
- Instruments XPTO3 and XPTO4 are in Group 10
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 is matched in 5 separate executions (partial fills)	New Order Fills counter increments to 1
2	A single resting order's ask in XPTO4 is matched in 3 separate executions	New Order Fills counter increments to 2
3	A single resting order's bid in XPTO4 is matched in 1 execution	New Order Fills counter increments to 3
4	A single resting order's ask in XPTO3 is matched in 2 separate executions	New Order Fills counter increments to 4
5	A single resting order's bid in XPTO3 is matched in 5 separate executions	New Order Fills counter increments to 5

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO3
- System attempts to cancel all remaining resting orders for Group 10
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to begin submitting orders to that group again

Note: Even though the first bid order was hit 5 times, it only equates to a counter value of '1' because no other order was entered by the customer or executed during the time interval. Similarly, even though the first ask was taken 3 times, it only equates to a counter value of "1" because only 1 order was entered by the client system and executed during the time interval.

14.7.1.2 Execution Protection

The customer specifies a threshold for the number of executions, or actual fills, on a Group during the Time Interval. As soon as the Execution Protection threshold is met or exceeded, the PUMA platform initiates protection of the customer's orders for instruments associated with the Group.

The PUMA platform allows this protection to be exceeded in instances where a single inbound order matches with several resting orders. The PUMA Trading System does not stop the match process during a single match event.

Note: If used in conjunction with the New Order Fill protection, this value must be set to a number equal to or greater than the New Order Fill Protection setting.

Example 1

Consider the situation described below:

- An Execution Protection value of 10 for Group 10
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 10
- Instruments XPTO3 and XPTO4 are in Group 10
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 for a quantity of 500 is matched in 5 separate executions (partial fills)	New Order Fills counter increments to 5
2	A single resting order's ask in XPTO4 for a quantity of 300 is matched in 3 separate executions	New Order Fills counter increments to 8
3	A single resting order's bid in XPTO4 for a quantity of 100 is matched in 1 execution	New Order Fills counter increments to 9
4	A single resting order's ask in XPTO3 for a quantity of 100 is matched in 1 execution	New Order Fills counter increments to 10

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO3
- System attempts to cancel all remaining resting orders for Group 10
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

Example 2: Triggered Stop Order Cancelled at Protection Mode Activation

Consider the situation described below:

- An Execution Protection value of 10 for Group 10
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 10, including a stop order for instrument XPTO3
- Instruments XPTO3 and XPTO4 are in Group 10
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 for a quantity of 500 is matched in 5 separate executions (partial fills)	New Order Fills counter increments to 5
2	A single resting order's ask in XPTO4 for a quantity of 300 is matched in 3 separate executions	New Order Fills counter increments to 8
3	A single resting order's bid in XPTO4 for a quantity of 100 is matched in 1 execution	New Order Fills counter increments to 9
4	A single resting order's ask in XPTO3 for a quantity of 100 is matched in 1 execution at a price good to trigger the stop order registered for the instrument	New Order Fills counter increments to 10

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO3
- System attempts to cancel all remaining resting orders for Group 10, including the stop order triggered after the last execution in XPTO3.
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

14.7.1.3 Traded Quantity Protection

The customer specifies a threshold for the number of traded quantities of contracts within a Group that can take place during the Time Interval.

As soon as the Traded Quantity Protection threshold is reached or exceeded, the PUMA Trading System initiates protection of the customer's orders for instruments associated with the Group.

The PUMA platform allows this protection to be exceeded in instances where a single inbound order matches with several resting orders. The PUMA Trading System does not stop the match process during a single match event.

Example 1

Consider the situation described below:

- A Traded Quantity Protection value of 1000 for Group 10
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 10
- Instruments XPTO3 and XPTO4 are in Group 10
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 for a quantity of 200 is matched in 2 separate executions (partial fills)	Traded Quantity Counter increments to 200
2	A single resting order's ask in XPTO3 for a quantity of 700 is matched in 7 separate executions	Traded Quantity Counter increments to 900
3	A single resting order's ask in XPTO4 for a quantity of 100 is matched in 1 execution	Traded Quantity Counter increments to 1000
4	Protection mode is enabled for Group 10.	All remaining resting orders for Group 10 are cancelled

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO4
- System attempts to cancel all remaining resting orders for Group 10
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

Example 2: Iceberg Order Filled and Protection Value Exceeded

Consider the situation described below:

- A Traded Quantity Protection value of 1000 for Group 10
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 10
- Instruments XPTO3 and XPTO 4 are in Group 10
- There is an iceberg order of 1000 (50000) shares for the instrument XPTO 3
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 for a quantity of 200 is matched in 2 separate executions (partial fills)	Traded Quantity Counter increments to 200
2	A single resting order's ask in XPTO3 for a quantity of 700 is matched in 7 separate executions	Traded Quantity Counter increments to 900
3	The iceberg order in XPTO3 for a quantity of 1000 (50000) is matched by a single order of 10000 shares in 1 execution	Traded Quantity Counter increments to 10900
4	Protection mode is enabled for Group 10.	All remaining resting orders for Group 10 are cancelled

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO3
- System attempts to cancel all remaining resting orders for Group 10
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

14.7.1.4 Buy/Sell Protection

The PUMA Trading System triggers Buy/Sell Protection when the absolute value of the Buy/Sell Protection parameter is greater than or equal to the value defined by the customer.

The Buy/Sell Protection parameter counts the number of contracts traded in a Group within the Time Interval.

When strategies are involved, the buys and sells are determined at the outright leg level for exchange-defined spreads and UDS. Buy trades increase the buy/sell count by one and Sell trades decrease the buy/sell count by one.

All instrument types (cash, futures, options) are counted equally.

Example

Consider the situation described below:

- A Buy/Sell Protection value of +/- 2500 (absolute value) for Group 10
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 10
- Instruments XPTO3 and XPTO4 are in Group 10
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's bid in XPTO4 for a quantity of 2000 is matched in 5 separate executions (partial fills)	Buy/Sell Protection = 2000
2	A single resting order's ask in XPTO3 for a quantity of 200 is matched in 2 separate executions	Buy/Sell Protection = ABS(+2000 + -200) = 1800
3	A single resting order's bid in XPTO4 for a quantity of 700 is matched in 1 execution	Buy/Sell Protection = ABS(+1800 + 700) = 2500
4	Protection mode is enabled for Group 10.	All remaining resting orders for Group 10 are cancelled

As a result:

- Protection mode would be enabled for Group 10 after the last match event ends in XPTO4
- System attempts to cancel all remaining resting orders for Group 10
- Any new incoming orders for Group 10 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

14.7.1.5 Delta Protection

Delta measures the rate of change of an option premium with respect to a price change in the underlying contract. Delta is a measure of price sensitivity at any given moment.

Not all options move point-for-point with their underlying futures contracts. If a futures contract moves .50 points and the option only moves .25 points, its delta is 50%; i.e., the option is only 50% as sensitive to the movement of underlying futures contract.

The delta will change as an option moves from out-of-the money to at-the-money to in-the-money, approaching 100%. Deltas range from 0% to 100%. The delta of the underlying futures contract is 100%.

The PUMA Trading System triggers Delta Protection when the absolute value of the Delta Protection parameter is greater than or equal to the value defined by the customer.

This protection assumes that all of the contracts in the Group have some delta defined.

Delta Protection compares a **Delta Counter Value (W)** with a **Delta Static Value**. If the absolute value of W increments or decrements to a value great than or equal to the Delta Static Value within **Time Interval (N)**, then the Delta Protection is triggered.

Delta	Description
Delta Static Value	It is a minimum/maximum delta protection value defined by the customer per Group. One value is assumed to be positive and negative, i.e. 300 means +300 and -300 deltas.
Delta Counter Value (W)	Increments and decrements deltas per Group per customer. • Buying calls and selling puts increment W by each contract's delta value. • Selling calls and buying puts decrement W by each contract's delta value.
Time Interval (N)	Resets W to zero every N seconds unless the protection is triggered.

Example

Consider the situation described below:

- Delta Static Value of +/- 60000 for Group 12
- Time Interval Value = 15 seconds
- A customer has multiple resting orders for Group 12
- Instruments XPTOA10, XPTOM10, XPTOA11 and XPTOM11 are in Group 12
- XPTOA10 Call Bid Delta value = +50
- XPTOA10 Call Ask Delta value = -50
- XPTOM10 Put Bid Delta value = - 60
- XPTOM10 Put Ask Delta value = +60
- XPTOA11 Call Bid Delta value = +55
- XPTOA11Call Ask Delta value = -55
- XPTOM11 Put Bid Delta value = - 65
- XPTOM11 Put Ask Delta value = +65
- No new orders are entered during the 15 seconds time interval

Within the 15-seconds time interval, the following events occur:

#	Event	Comment
1	A single resting order's Call bid in XPTOA10 for a quantity of 1000 is matched in 2 separate executions (partial fills)	Delta Counter Value = 0 + (+50 x 1000) = 50000
2	A single resting order's Call ask in XPTOA10 for a quantity of 200 is matched in 1 execution	Delta Counter Value = 50000 + (-50 x 200) = 40000
3	A single resting order's Put ask in XPTOM11 for a quantity of 400 is matched in 1 execution	Delta Counter Value = 40000 + (+65 x 400) = 66000
4	Protection mode is enabled for Group 12.	All remaining resting orders for Group 12 are cancelled

As a result:

- Protection mode would be enabled for Group 12 after the last match event ends in XPTOM11
- System attempts to cancel all remaining resting orders for Group 12
- Any new incoming orders for Group 12 without reset tag would be rejected
- The customer will have to send a reset message to being sending orders to that group again

14.7.2 Protection Counters

It's important to observe that the Market Protections counters might diverge in the way they are incremented depending on the type of operation been executed.

14.7.2.1 Cross Order

A direct operation registered via message NewOrderCross (35=s) will affect the Market Protections counters as if there were two separated executions.

14.7.2.2 User Defined Spreads (UDS)

Counters of the Delta and Buy/Sell Protections consider the executions on each leg of the strategies. On the other hand, New Order Fill, Execution and Traded Quantity Protections are only concerned to the execution of the UDS itself.

14.7.3 Automatic Reset

There are some specific circumstances where Market Protections values are automatically reset by the PUMA Trading System.

14.7.3.1 Next Trading Session

When trading session is closed for the day, the Market Protections are reset. During the next trading day, the instrument groups will start in the Monitoring Mode accepting new orders normally, even if the Protection Mode was enabled at the end of the previous day.

14.7.3.2 Template update

Market Protections have a set of pre-configured parameters provided to the participants as templates. When an update is performed on a template, all sessions related to that template will have their Market Protections reset.

14.7.4 FIX Tags Usage

EntryPoint supports the Market Protections functionality by providing the set of tags and error codes that allow the trading platform to communicate with the clients and inform them about the events triggered by the functionality.

Additionally, the order entry interface allows the clients to reset the Monitoring Mode once they are ready to trade again. Find below the changes made to the order entry interface.

14.7.4.1 Protected Mode

In Protected Mode, the trading platform will cancel remaining orders and prevent the entry of new orders for all instruments associated with the protected group.

After canceling the orders, the trading platform sends Execution Reports (35=8) of cancelation to the participant with tags ExecType (150) = 4 and ExecRestatementReason (378) = 200.

Tag	Tag name	Req'd	Data Type	Comment
378	ExecRestatementReason	N	Int (6)	Indicates reason of restatement, if available. Valid values: 8 - Market Option 100 - Cancel On Hard Disconnection 101 - Cancel On Logout 102 - Cancel On Disconnect And Logout 103 - Self Trading Prevention 200 - Market Protections

14.7.4.2 Resetting Monitoring Mode

Once Market Protections are triggered, the Exchange will not accept new orders from the Market Maker for that product group.

When the client system is ready to re-submit orders, it is necessary to notify the PUMA Trading System to restart the Monitoring Mode by sending tag MMProtectionReset (9773) = Y in the New Order Single (35=D) message.

Although modifications of existing orders do not require the use of tag MMProtectionReset (9773), one may include this tag in the OrderCancelReplaceRequest (35=G) message in order to reset the monitoring mode.

This tag makes the platform to accept new orders for the protected group again.

Tag	Tag name	Req'd	Data Type	Comment
9773	MMProtectionReset	N	Boolean (1)	Resets the Market Protections. Valid value: Y - Reset Market Protections

14.7.4.3 Rejection Message

After Protected Mode is triggered, any new order addressed to that product group, without tag MMProtectionReset (9773) = Y, will be refused with an Execution Report (35=8) of rejection with tag ExecType (150) = 8.

Tags OrdRejReason (103) = 2600 and Text (58) = “Market Protections in effect for user <SenderCompId> and group <Group>” will help to identify the cause of rejection.

15. Application Message Scenarios

The following sections provide examples of the most common application message scenarios. In all scenarios, if a message is malformed or fails specific business level conditions, it will be rejected with either a Session Reject (invalid tag for message, invalid body length, etc) or Business Message Reject message (e.g., conditionally required field missing).

15.1 Order Management

15.1.1 Order Entry, Partial Fill and Complete Fill

In this example, an order is sent by the client institution. This order is partially filled and is completely filled afterwards.

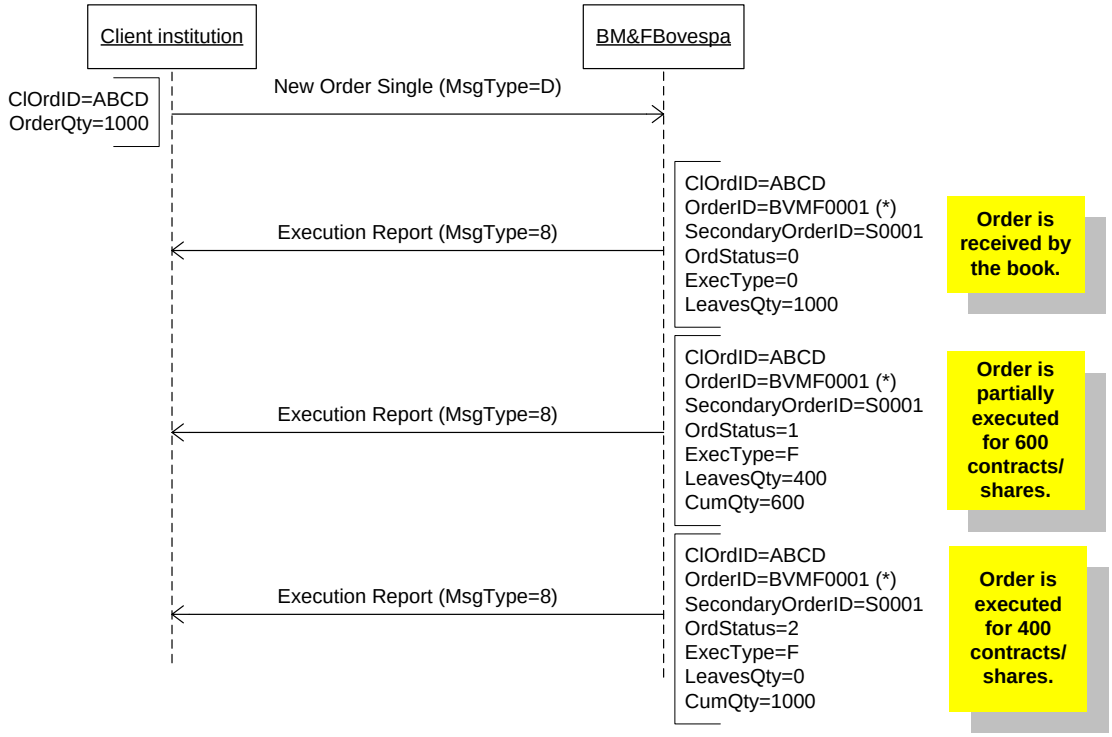

Figure 7 - Order Entry with partial and total fill

15.1.2 **Order Cancelation by ClOrdID**

In this example, the client institution issues an order, and cancels it afterwards referring to its ClOrdID. The ClOrdID was generated by the issuer of the order, and must be unique for that FIX session and instrument. BM&FBOVESPA correlates the ClOrdID issued by the client with its own internal order ID per instrument, sent to the client in the tag OrderID in the Execution Report messages.

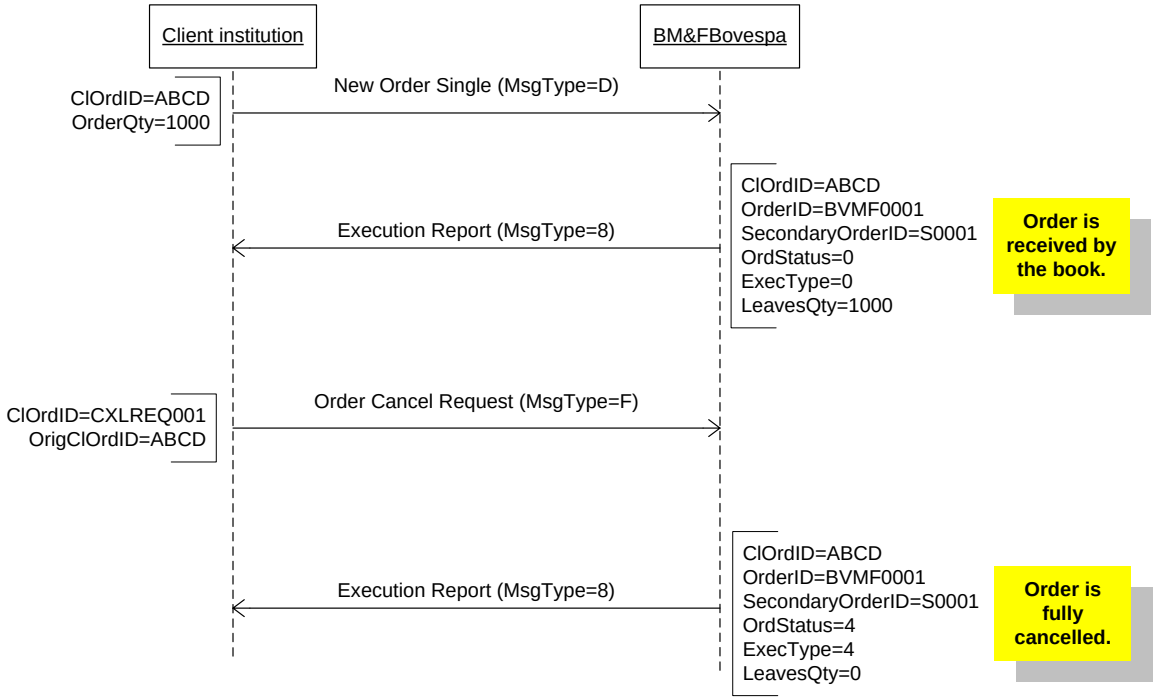

Figure 8 - Order cancelation using ClOrdID

15.1.3 **Order Cancelation by OrderID**

Once an order is accepted by BM&FBOVESPA, it is assigned a unique internal identifier by instrument, sent to the client in the tag OrderID in each Execution Report message. The client may take action on that order using the OrderID instead of the OrigClOrdID.

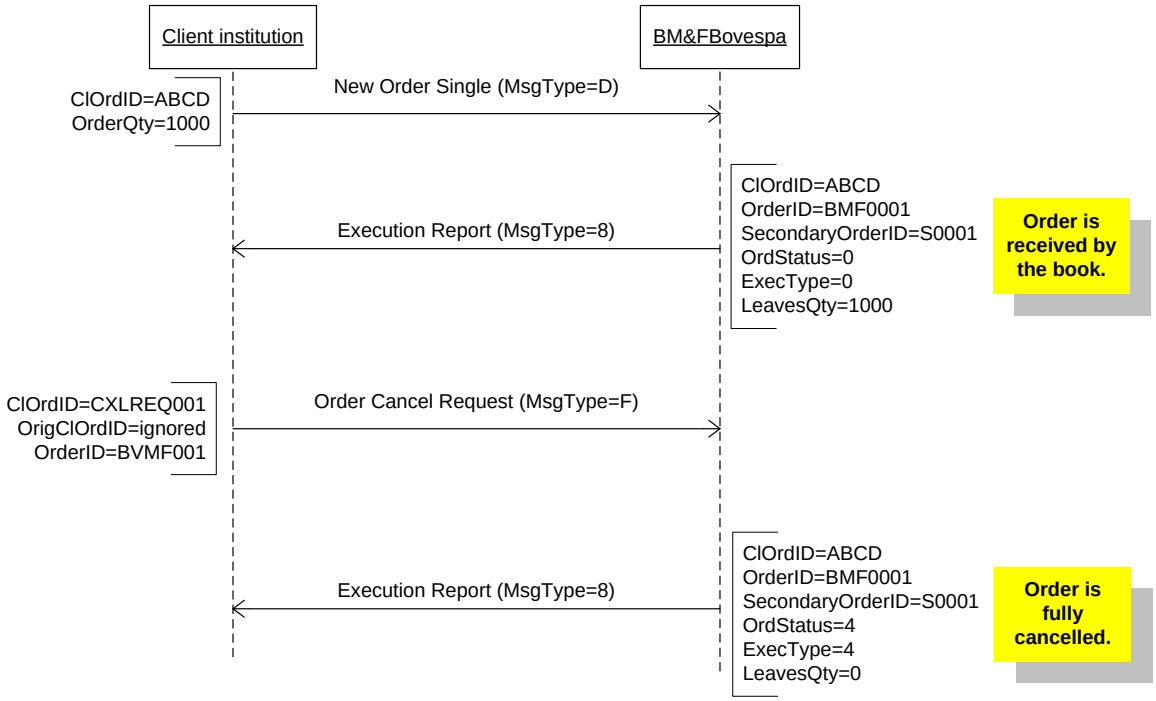

Figure 9 - Order cancelation by OrderID

15.1.4 **Order Cancelation Attempt of Filled Order**

In this example, the client issues a new order, this order is filled, and the client attempts to cancel the filled order. The cancel request will be rejected.

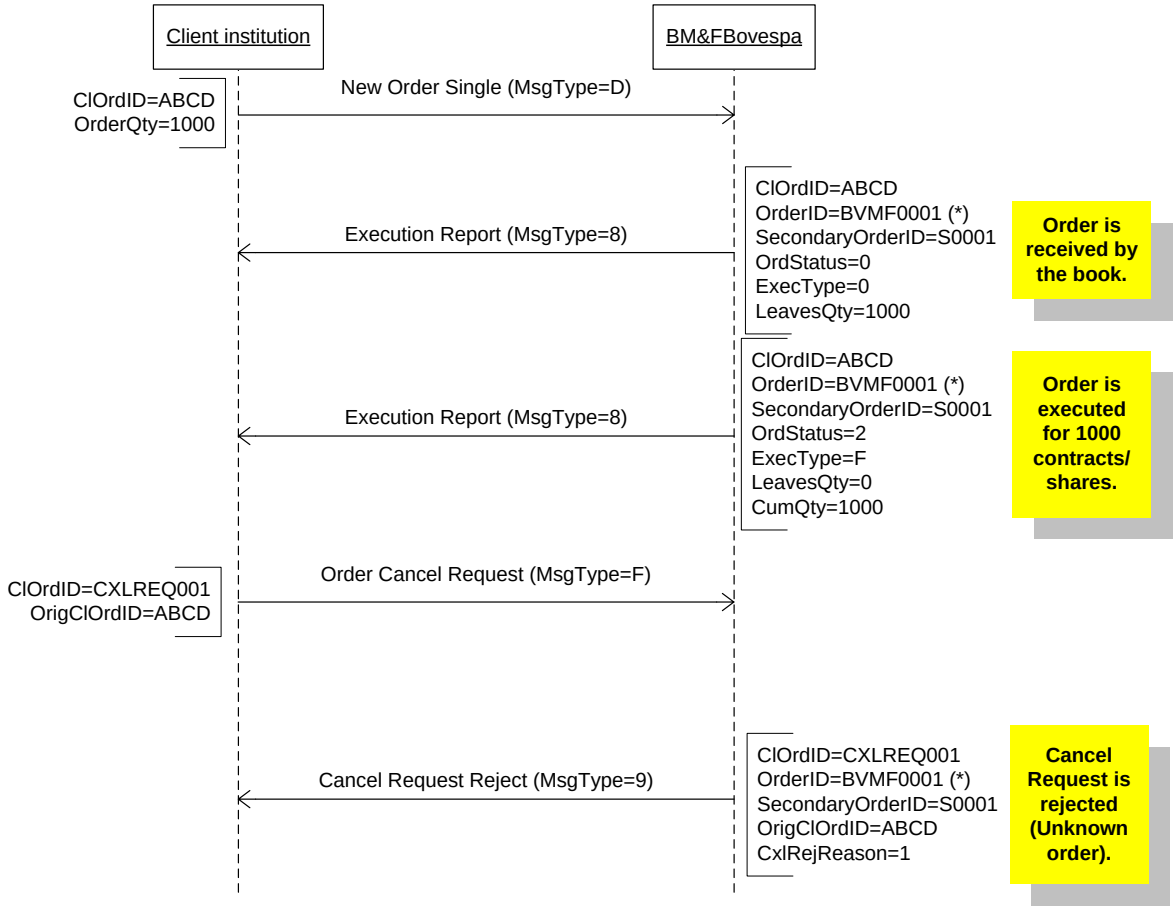

Figure 10 - Attempt to cancel a filled order

15.1.5 **Order Modification**

This example illustrates the modification of an order issued by the client. Notice that an order that is modified keeps the BM&FBOVESPA order ID (OrderID) of the cancelled order.

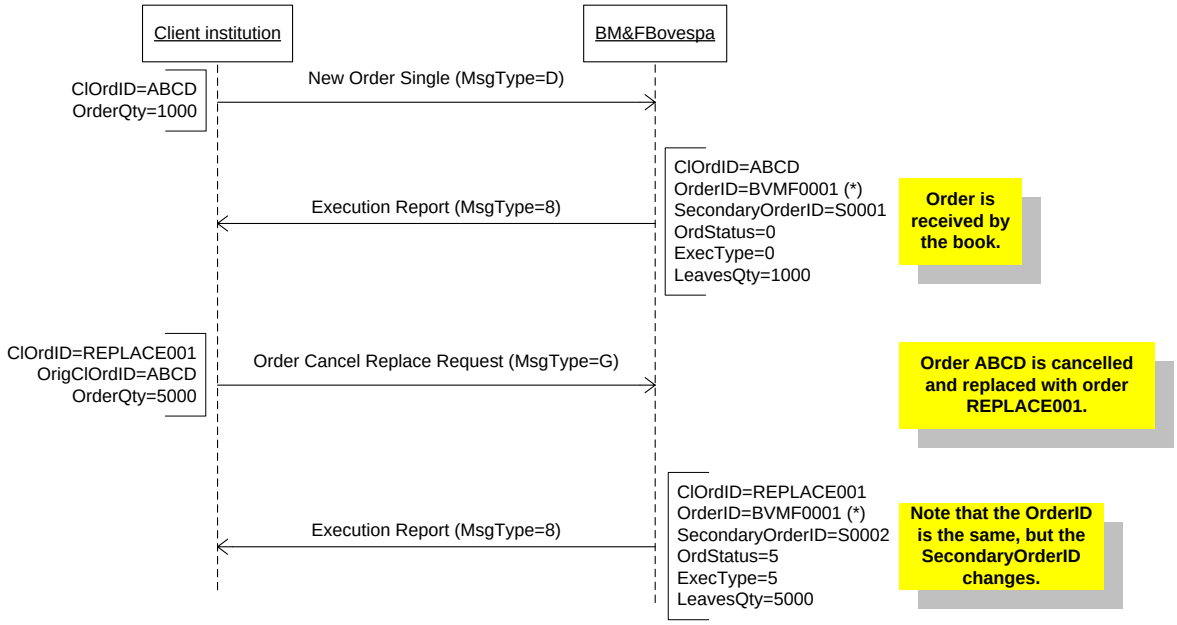

Figure 11 - Order modification scenario - OrderID is kept for modified order.

15.1.6 **Cross Order**

The New Cross Order message (tag 35=s) is used by institutions to electronically submit orders to buy and sell the same security for different investors through the register of direct operation in the trading system.

In EntryPoint, the use of cross orders is available not only to desk traders, but to all participants, independently the type of access used to connect to the Exchange. Some restrictions apply for sessions making use of the LiNe credit control system. Such scenario must be evaluated during the customer's system certification process.

The acknowledgment of receipt of a New Cross Order message is issued by BM&FBOVESPA in the form of two Execution Report messages (tag 35=8). The order may be accepted (tag 150=0) or rejected (tag 150=8) according to BM&FBOVESPA rules.

If the cross trading meets any of parameters determined for cross trade auctions, the security will be submitted to a regular auction. If there are any valid offers at better prices (buying or selling) the cross order will be reject.

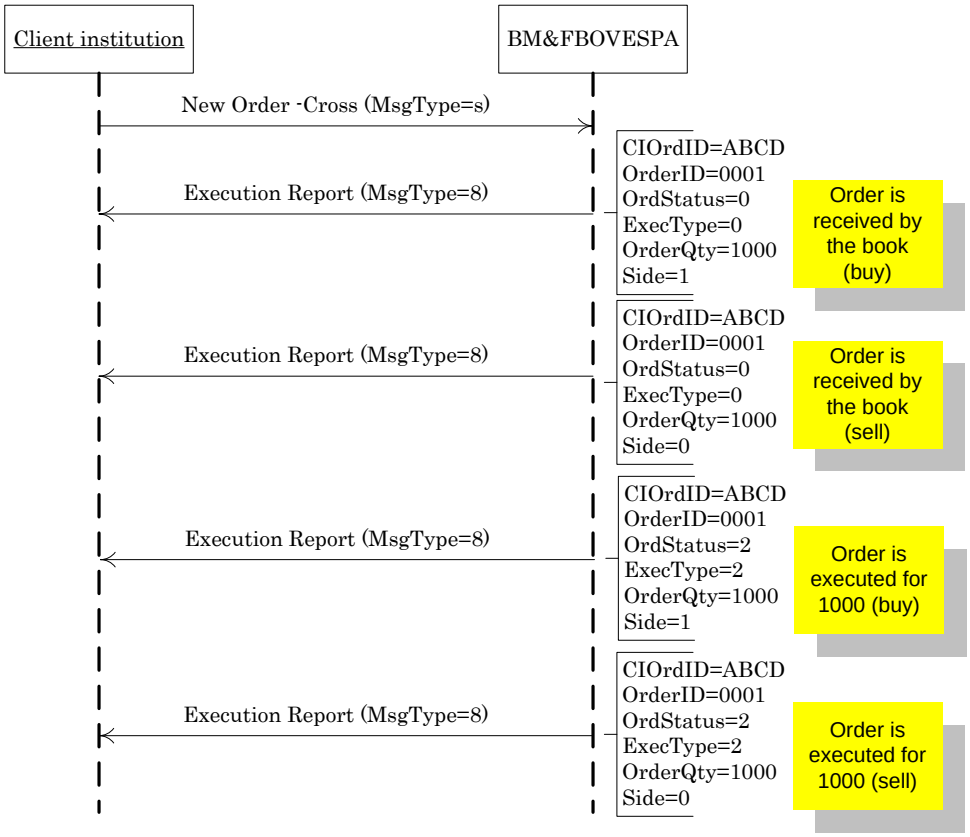

Figure 12 - New Order Cross scenario

15.2 Cancel On Disconnect

15.2.1 COD Disabled

In the example below, the user logs on to the Border Gateway with CancelOnDisconnectType set as “Do Not Cancel on Disconnect or Logout” (35002=0) and CODTimeoutWindow set as 0 seconds (35003=0). Later, the user logs out and the Border Gateway does not send any request for cancelation to Trading System. User logs back onto Border Gateway with CancelOnDisconnectType set as “Do Not Cancel on Disconnect or Logout” (35002=0) and CODTimeoutWindow set as 0 seconds (35003=0). At some point, the system detects a disconnection and the Border Gateway does not send any request for cancelation to Trading System.

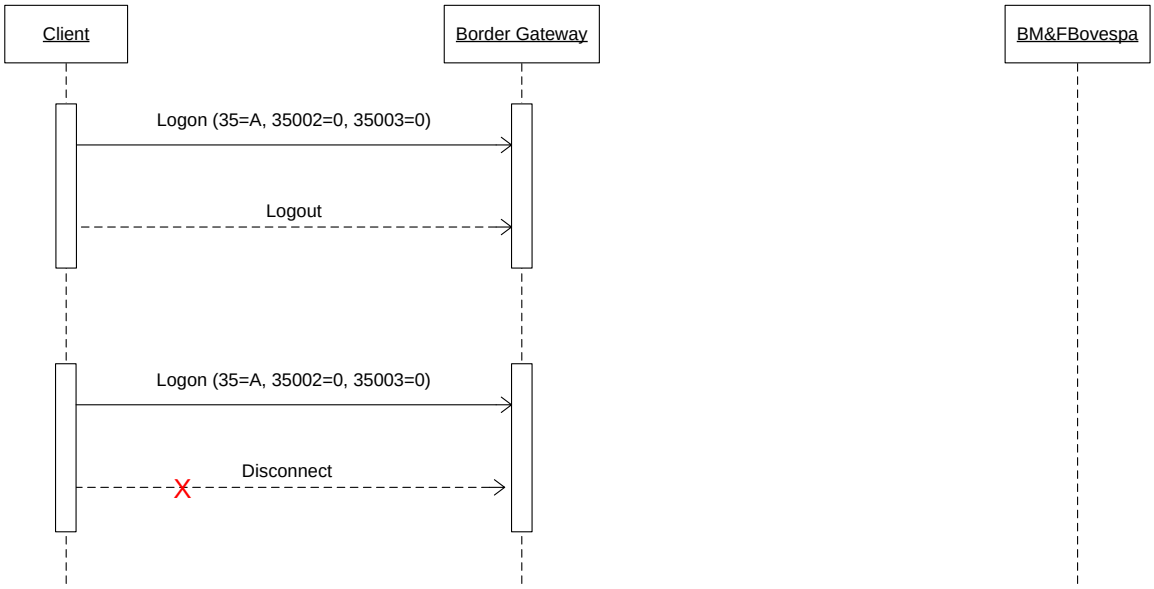

Figure 13 - Do Not Cancel On Disconnect or Logout

15.2.2 **Cancel On Disconnect Only**

In the next example, the user logs on to the Border Gateway with CancelOnDisconnectType set as “Cancel on Disconnect only” (35002=1) and CODTimeoutWindow set as 30 seconds (35003=30000). The user voluntarily logs out from the system and COD is not triggered because the user chose to cancel on disconnect only.

Then, the user logs back in using the same COD parameters and later the Border Gateway detects a disconnection. The user is not able to reconnect during the next 30 seconds. Border Gateway then sends a request for cancelation of all user non-GT orders on Trading System. As a result, two resting orders are cancelled and the respective Execution Reports are sent to the Border Gateway.

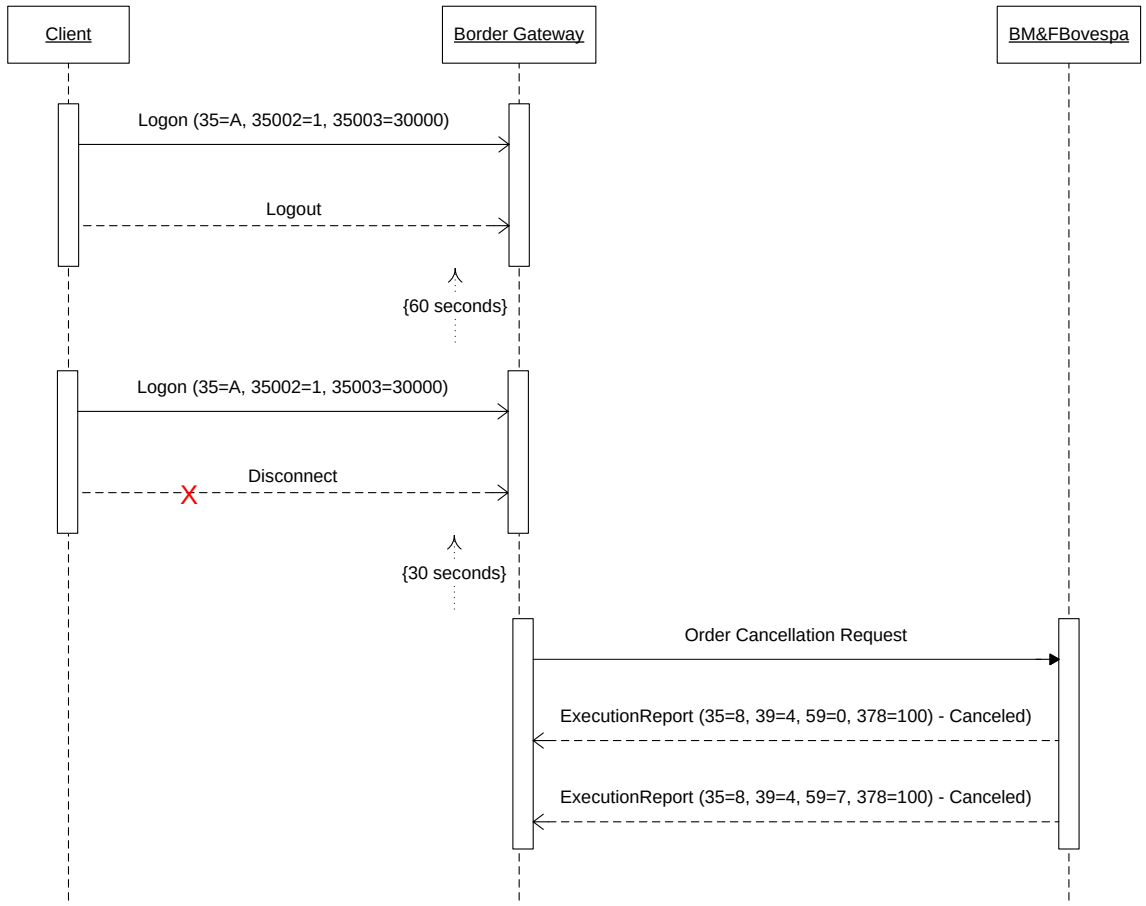

Figure 14 - Cancel On Disconnect Only

15.2.3 **Cancel On Logout Only**

In the example below, the user logs on to the Border Gateway with CancelOnDisconnectType set as “Cancel on Logout Only” (35002=2) and CODTimeoutWindow set as 30 seconds (35003=30000). At some point, the Border Gateway detects a disconnection. The system doesn’t trigger the order cancelations given that the user chose to cancel on logout only.

One minute later, the user logs back in using the same COD parameters. The user voluntarily logs out and does not reconnect within the next 30 seconds. Border Gateway then sends a request for cancelation of all user non-GT orders on Trading System. As a result, two resting orders are cancelled and the respective Execution Reports are sent to the Border Gateway.

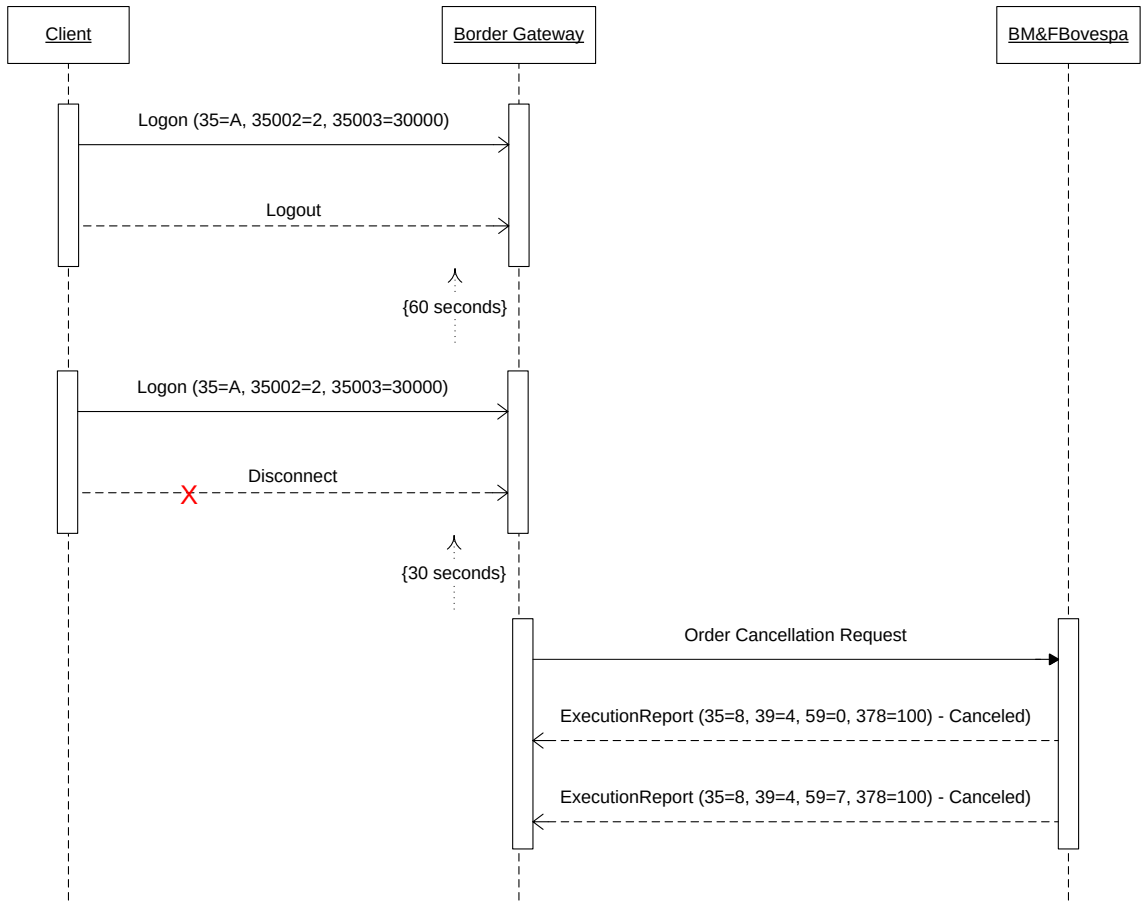

Figure 15 - Cancel On Logout Only

15.2.4 **User Logs Back In before COD Timeout Window elapses**

In this example, the user logs on to the Border Gateway with CancelOnDisconnectType set as “Cancel on Disconnect Only” (35002=1) and CODTimeoutWindow set as 30 seconds (35003=30000).

At some point, the Border Gateway detects a disconnection and the COD starts the countdown to trigger the order cancelations as soon as the timeout expires. Meanwhile, the user reconnects before the timeout expires and, as a result, the Border Gateway does not send any cancelation request to Trading System.

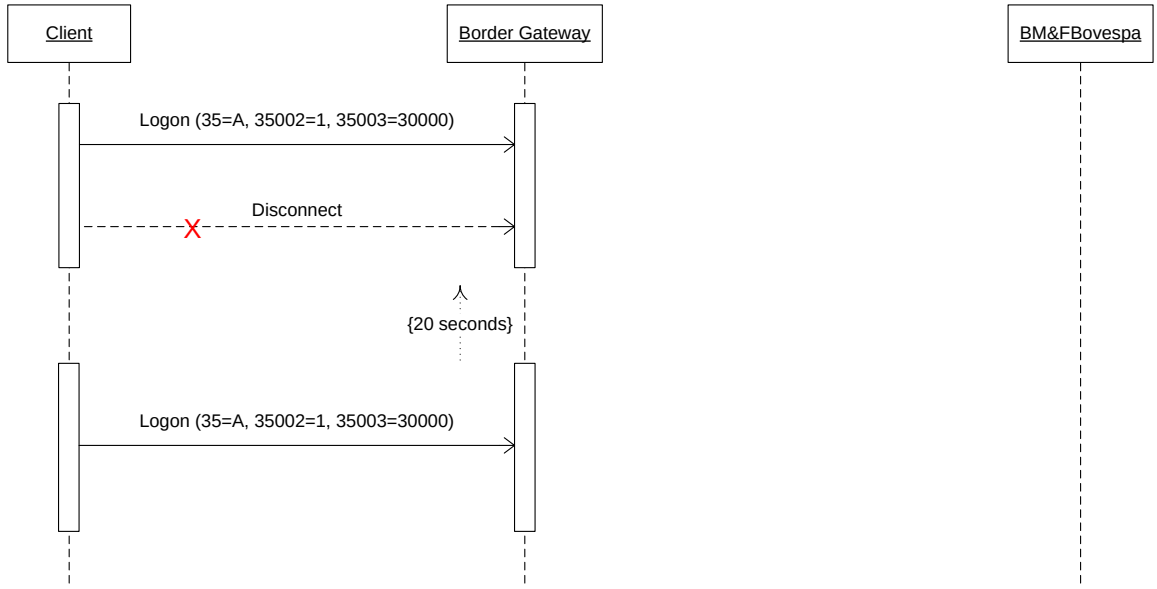

Figure 16 - User Logs Back In

15.3 Exercise & Blocking

15.3.1 Options Exercise

In this example, the client submits a Position Maintenance Request in order to exercise an equity options contract position. BM&FBOVESPA processes the request and, if the Position Maintenance Request is processed successfully, it sends to the client a Position Maintenance Report with tag PosMaintStatus (722=3 Completed).

Additionally, BM&FBOVESPA sends one or more Execution Reports (35=8) to the client and all counterparties involved in the transaction. Each Execution Report (35=8) contains tag OrderCategory (1115) with the proper value according to the side that is exercising its position (1115=B) and the side that is being assigned by the Exercise (1115=C).

As a position can be exercised against many counterparties, it's important to note that a single Exercise request may generate many Execution Reports that will be sent also to the counterparties involved in the transaction.

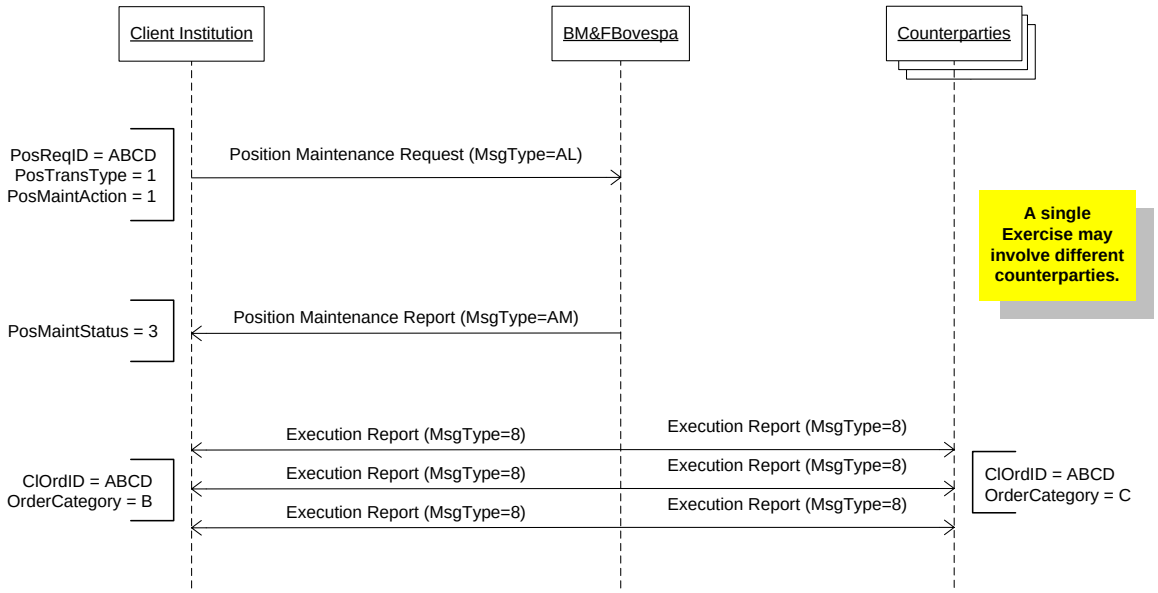

Figure 17 - Option exercise scenario.

15.3.2 **Automatic Blocking**

When an equity option's purchase occurs in the matching engine, BM&FBOVESPA checks whether the client has a position on the same options contract that can be blocked by the purchase and updates client's position.

If the operation resulted in a blocked position, BM&FBOVESPA sends a Position Maintenance Report to notify the client.

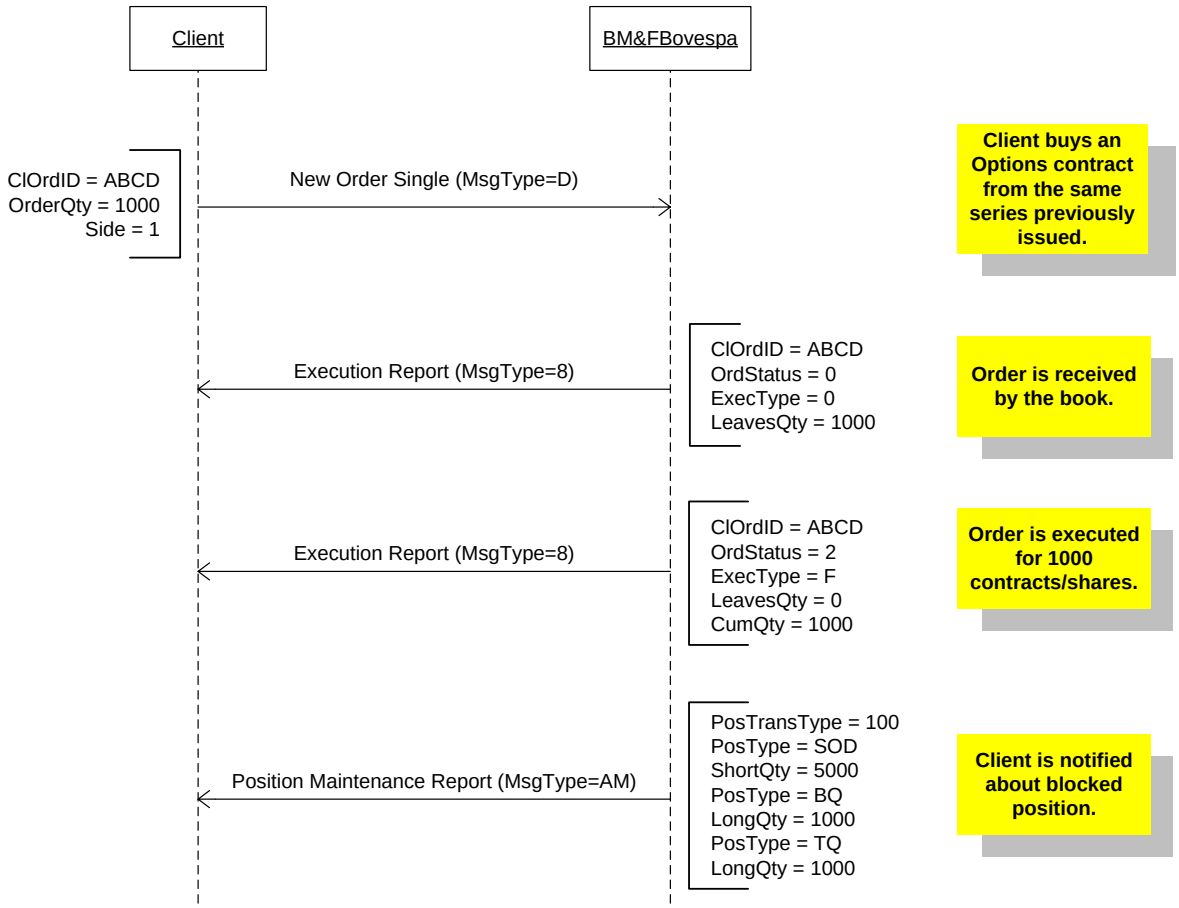

Figure 18 - Automatic Blocking scenario

15.3.3 **Blocking Specification**

This example illustrates a client wishing to Block an equity options contract position, i.e. to prevent a position of being assigned. Client submits an Allocation Instruction in order to Block an options contract position.

BM&FBOVESPA processes the request and, in case the Allocation Instruction is processed successfully, client will receive an Allocation Report confirming the operation.

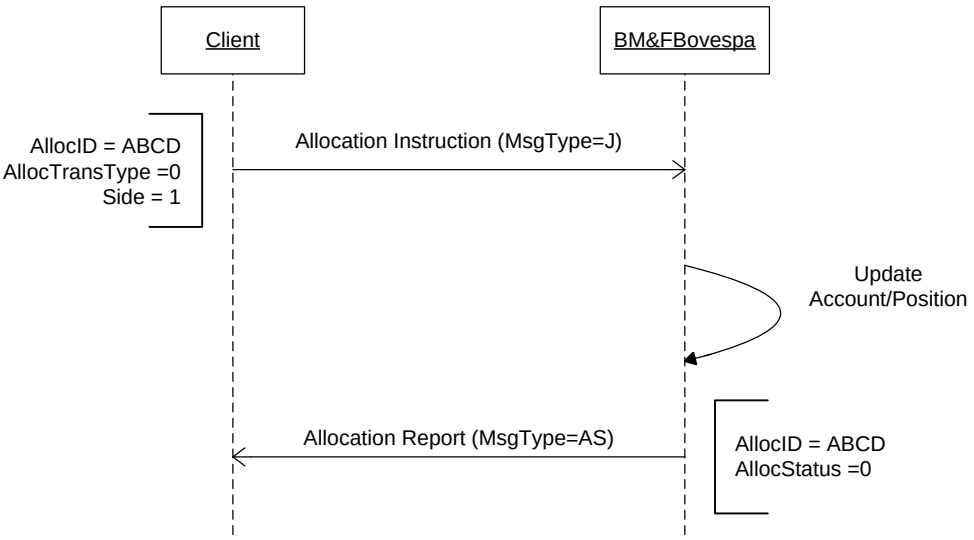

Figure 19 - Blocking Specification scenario

15.4 Scheduled Exercise on Stock and Equity EFT's Options

15.4.1 Scheduled Exercise Request

Participants will be able to request the Scheduled Exercise only at the maturity date of the options contract.

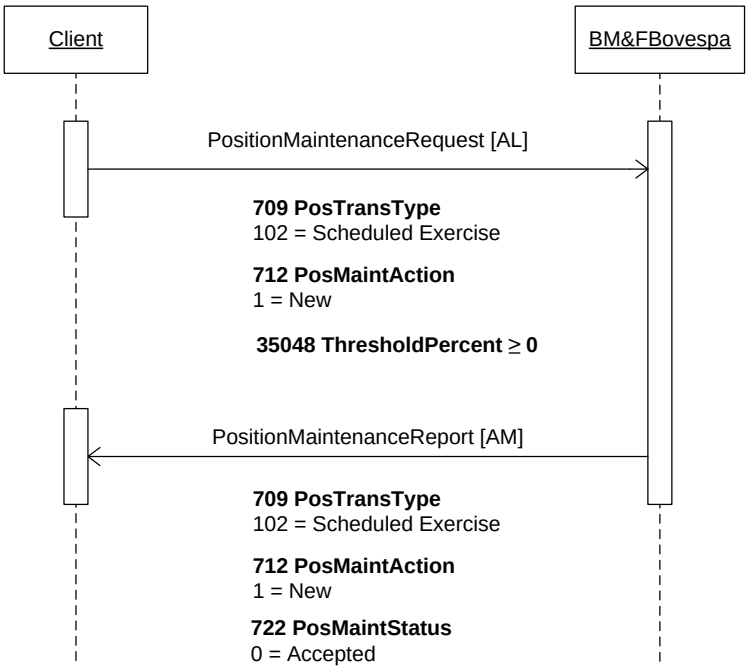

Figure 20 – Scheduled Exercise Request Accepted

1. In order to register the Scheduled Exercise for an Options contract, the Holder submits a Position Maintenance Request (35=AL) message with tag PosTransType (709=102 Scheduled Exercise), tag PosMaintAction (712=1 New) and a value in tag ThresholdPercent (35048) indicating the minimum percentage in the money required for the series to be exercised.
2. The system processes the request.
3. If the Position Maintenance Request (35=AL) message is processed successfully, BM&FBovespa sends to the Holder a Position Maintenance Report (35=AM) message with tag PosMaintStatus (722=0 Accepted).

15.4.2 Rejection of Scheduled Exercise Request

In case the request is rejected, tags PosMaintResult (723) and Text (58) will respectively inform the code and description of the cause of the rejection message.

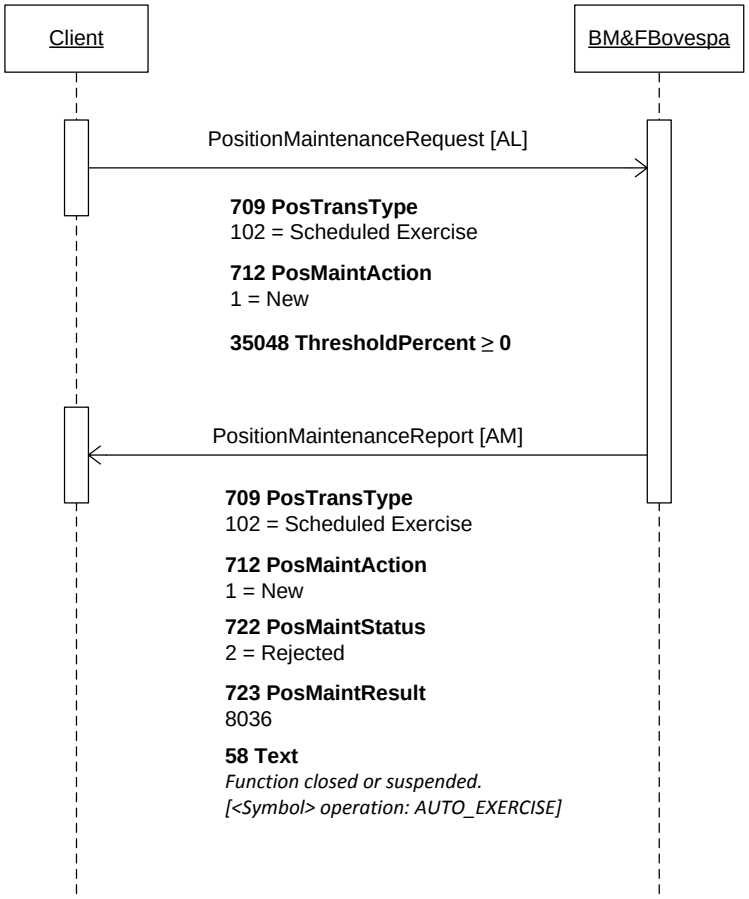

Figure 21 – Scheduled Exercise Request Rejected

15.4.2.1 Invalid Quantity – Error Code 8005

In case the quantity provided in the request is greater than the quantity available for exercise, the request will be rejected and tag PosMaintResult (723) will be assigned with code 8005.

The description of the rejection in tag Text (58) will present the following message:

- *Requested exercise quantity is more than the available quantity [Requested <Requested Qty>; Available (Holder): <Available Qty on Holder>]*

15.4.2.2 Functionality Not Available – Error Code 8036

Should the Scheduled Exercise functionality not be available, the request will be rejected and tag PosMaintResult (723) will be assigned with code 8036.

The description of the error in tag Text (58) will present the following message:

- *Function closed or suspended. [<Symbol> operation: AUTO_EXERCISE]*

15.4.2.3 Invalid Threshold Percent – Error Code 8038

The value provided in tag ThresholdPercent (35048) must be a decimal number between 0 and 1, otherwise the request will be rejected and tag PosMaintResult (723) will be assigned with code 8038.

The description of the rejection in tag Text (58) will present the following message:

- *Threshold Percentage must be between 0 and 1.*

15.4.2.4 Threshold Percent not Provided – Error Code 8041

In case tag ThresholdPercent (35048) is not provided in the Scheduled Exercise request, the message will be rejected and tag PosMaintResult (723) will be assigned with code 8041.

The description of the rejection in tag Text (58) will present the following message:

- *Threshold must be informed in tag ThresholdPercent for Scheduled Exercise requests.*

15.4.2.5 Threshold Amount vs Threshold Percent – Error Code 8042

The use of tag ThresholdPercent (35048) is restricted to Scheduled Exercise requests. In case tag ThresholdPercent (35048) is present in a manual exercise request, the message will be rejected and tag PosMaintResult (723) will be assigned with code 8042.

The description of the rejection in tag Text (58) will present the following message:

- *Threshold must be informed in tag ThresholdAmount for manual exercises.*

15.4.3 Scheduled Exercise Request Cancellation by PosReqID

Participants will be able to request the cancellation of an Scheduled Exercise previously requested. In the cancellation message, participants need to inform the identifier of the request being cancelled.

In this case, tag OrigPosReqRefID (713) should be used to inform the original request id provided by the participant in tag PosReqID (710).

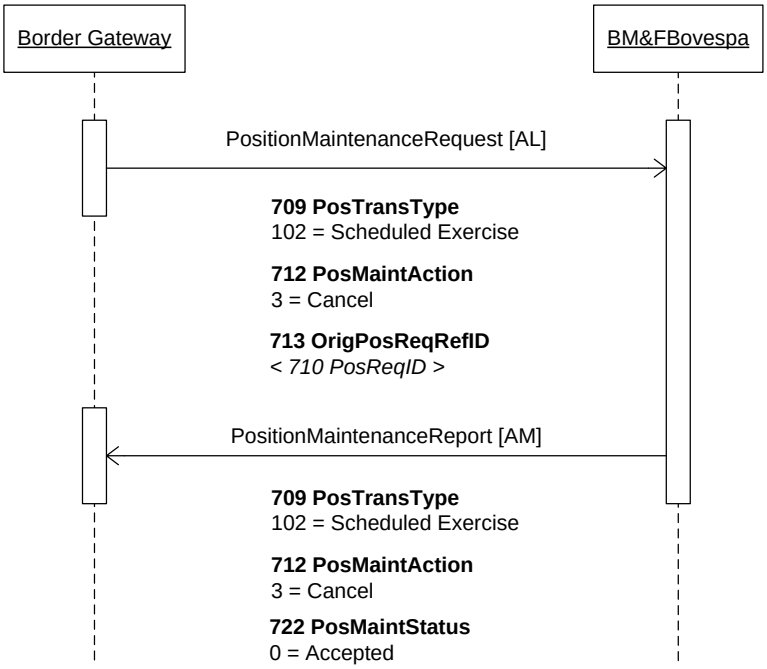

Figure 22 – Scheduled Exercise Request Cancellation by PosReqID

1. In order to register the Scheduled Exercise for an Options contract, the Holder submits a Position Maintenance Request (35=AL) message with tag PosTransType (709=102 Auto Exercise), tag PosMaintAction (712=3 Cancel) and inform in tag OrigPosReqRefID (713) the original id provided by the participant in the previous Scheduled Exercise Request.
2. The system processes the request.
3. If the Position Maintenance Request (35=AL) message is processed successfully, BM&FBovespa sends to the Holder a Position Maintenance Report (35=AM) message with tag PosMaintStatus (722=0 Accepted).

15.4.4 Scheduled Exercise Request Cancellation by PostMaintReptID

Alternatively, the participant might cancel the Scheduled Exercise request informing the identifier assigned by the Exchange in tag PostMaintReptID (721).

This identifier must be provided in tag PosMaintReptRefID (714) in the cancellation request.

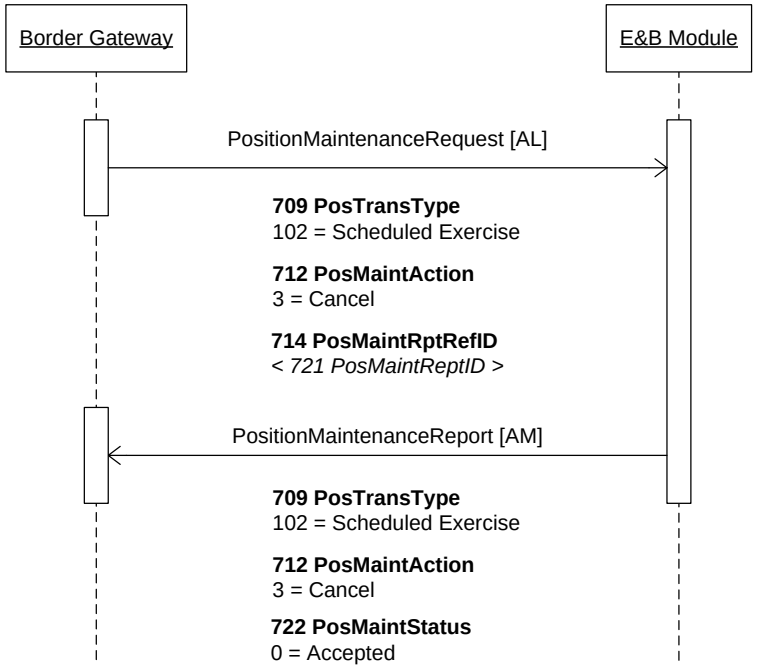

Figure 23 – Scheduled Exercise Request Cancellation by PostMaintReptID

1. In order to register the Scheduled Exercise for an Options contract, the Holder submits a Position Maintenance Request (35=AL) message with tag PosTransType (709=102 Auto Exercise), tag PosMaintAction (712=3 Cancel) and inform in tag PosMaintReptRefID (714) the original id provided by the Exchange when the previous Scheduled Exercise Request was accepted.
2. The system processes the request.
3. If the Position Maintenance Request (35=AL) message is processed successfully, BM&FBovespa sends to the Holder a Position Maintenance Report (35=AM) message with tag PosMaintStatus (722=0 Accepted).

15.4.5 Rejection of Scheduled Exercise Request Cancellation

In case the cancellation request is rejected, tags PosMaintResult (723) and Text (58) will respectively inform the code and description for the cause of the rejection message.

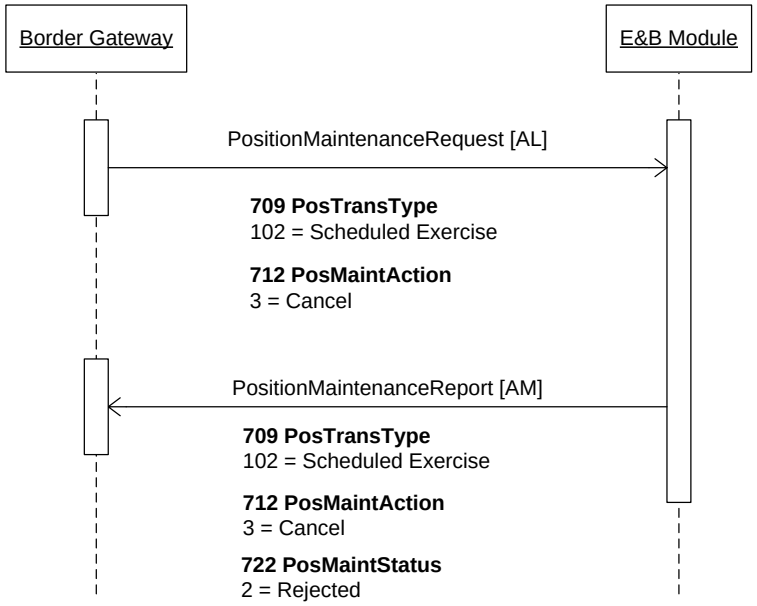

Figure 24 – Rejection of Scheduled Exercise Request Cancellation

15.4.5.1 Invalid Identifier – Error Code 8040

In case the identifier provided is incorrect, tag PosMaintResult (723) will be assigned with code 8040.

Tag Text (58) will convey one of the following messages, depending on the type of identifier provided PosReqID or PosMaintRptRefID:

- *Invalid exercise request Id to cancel [OrigPosReqRefID <request id> does not exist or does not belong to firm]*
- *Invalid exercise request Id to cancel [PosMaintRptRefID <request id> does not exist or does not belong to firm]*

15.4.5.2 Invalid Quantity – Error Code 8008

In case the quantity provided in the cancellation request does not match the quantity of the original request, the cancellation will be rejected and tag PosMaintResult (723) will be assigned with code 8008.

The description of the rejection in tag Text (58) will present the following message:

- *Invalid exercise quantity to cancel [Quantity on original request: <quantity on original request>].*

15.4.6 **Scheduled Exercise Notification**

When an Scheduled Exercise registered in the platform is executed, the system send messages notifying all entities involved in the transaction.

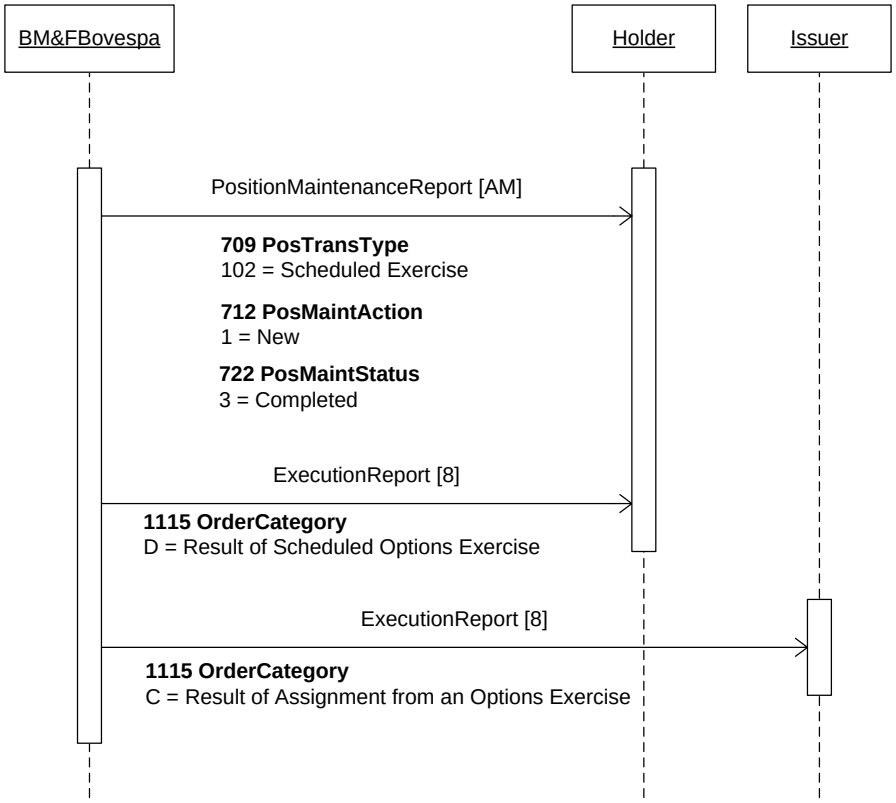

Figure 25 – Scheduled Exercise Notification

1. When the Scheduled Exercise is triggered, a PositionMaintenanceReport (35=AM) is sent to the Holder.
2. The ExecutionReport (35=8) sent to the Holder is identified with tag OrderCategory (1115) = D – identifying that the message is a result of a Scheduled Options Exercise.
3. The ExecutionReport (35=8) sent to the Issuer with tag OrderCategory (1115) = C – identifying that the message is a result of an assignment from an Options Exercise.

15.4.7 Scheduled Exercise Expiration

In case the Scheduled Exercise request is expired, a PositionMaintenanceReport (35=AM) message will be sent to the Holder with tag PosMaintStatus (722) = 9 – Not Executed.

Tags PosMaintResult (723) and Text (58) will respectively inform the code and description for the cause of the expiration message.

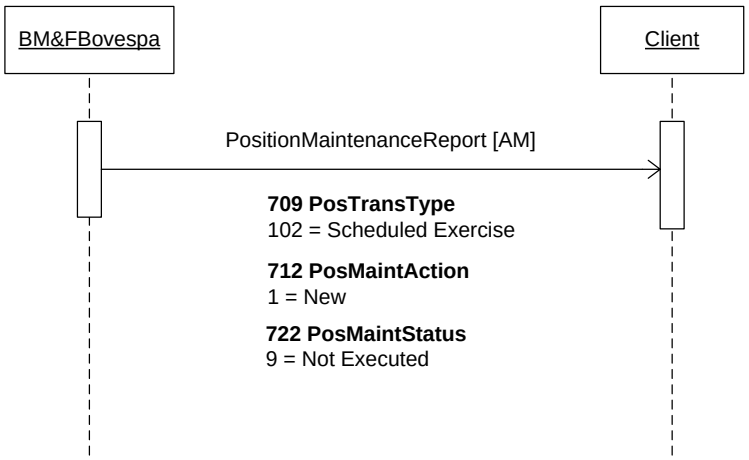

Figure 26 – Scheduled Exercise Expiration

15.4.7.1 Exercise Price not Allowed – Error Code 8006

In case the reference price does not match the required price, tag PosMaintResult (723) will be assigned with code 8006.

The description of the expiration in tag Text (58) will present the following message:

- *Exercise price not allowed [Referential price calculated: XX.XX, Minimum price requested to exercise automatically: XX.XX] (CALL) ou Exercise price not allowed [Referential price calculated: XX.XX, Maximum price requested to exercise automatically:XX.XX] (PUT)*

15.4.7.2 Unavailable Position – Error Code 8022

In case there's no position available to exercise, tag PosMaintResult (723) will be assigned with code 8022.

The description of the expiration in tag Text (58) will present the following message:

- *There aren't positions available to exercise [Requested: <Request quantity>; Available (Holder): 0]*

15.5 User-Defined Strategy

15.5.1 UDS Creation

This example illustrates the creation of a new instrument by the client. Client submits a Security Definition Request.

If the Security Definition Request is accepted, BM&FBOVESPA sends the client a Security Definition with tag Security Response Type (323 = 1 Accept security proposal as-is).

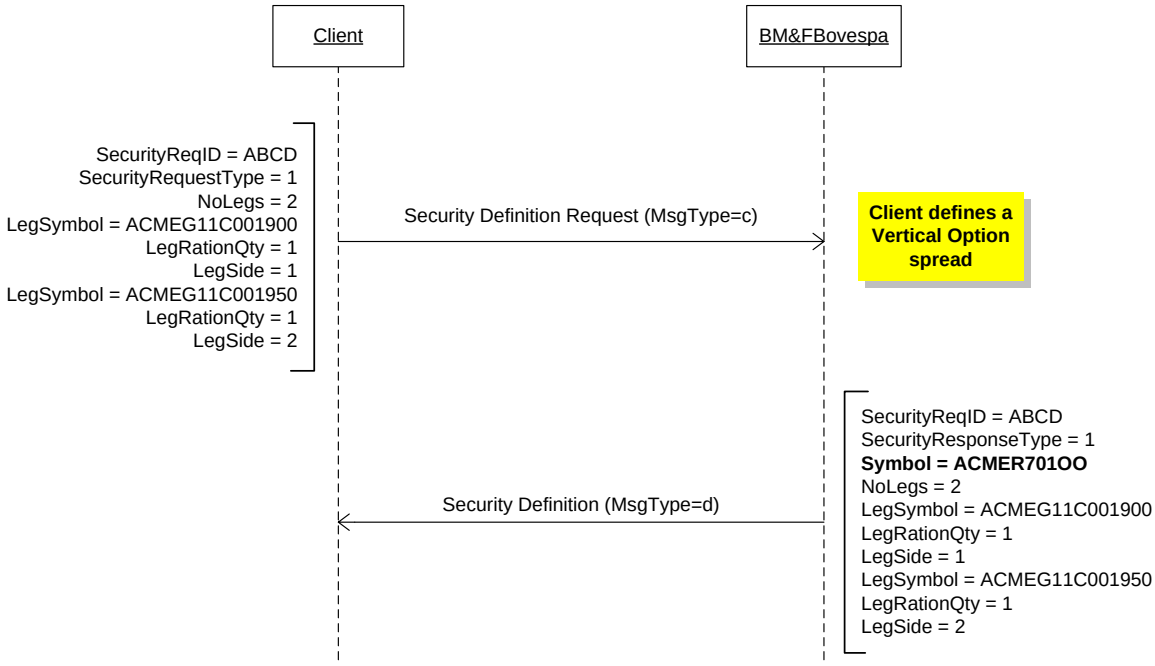

Figure 27 - User-Defined Spread creation scenario

15.5.2 **UDS Execution Report**

This example illustrates the flow of messages generated when a UDS is traded. The client places a New Order Single using the instrument previously created. BM&FBOVESPA sends a single Execution Report to acknowledge the receipt of the order.

When a match occurs, BM&FBOVESPA sends an Execution Report notifying the UDS' order fill (MultilegReportingType = 3). In addition, it sends Execution Reports for the fills of each leg (MultilegReportingType = 2).

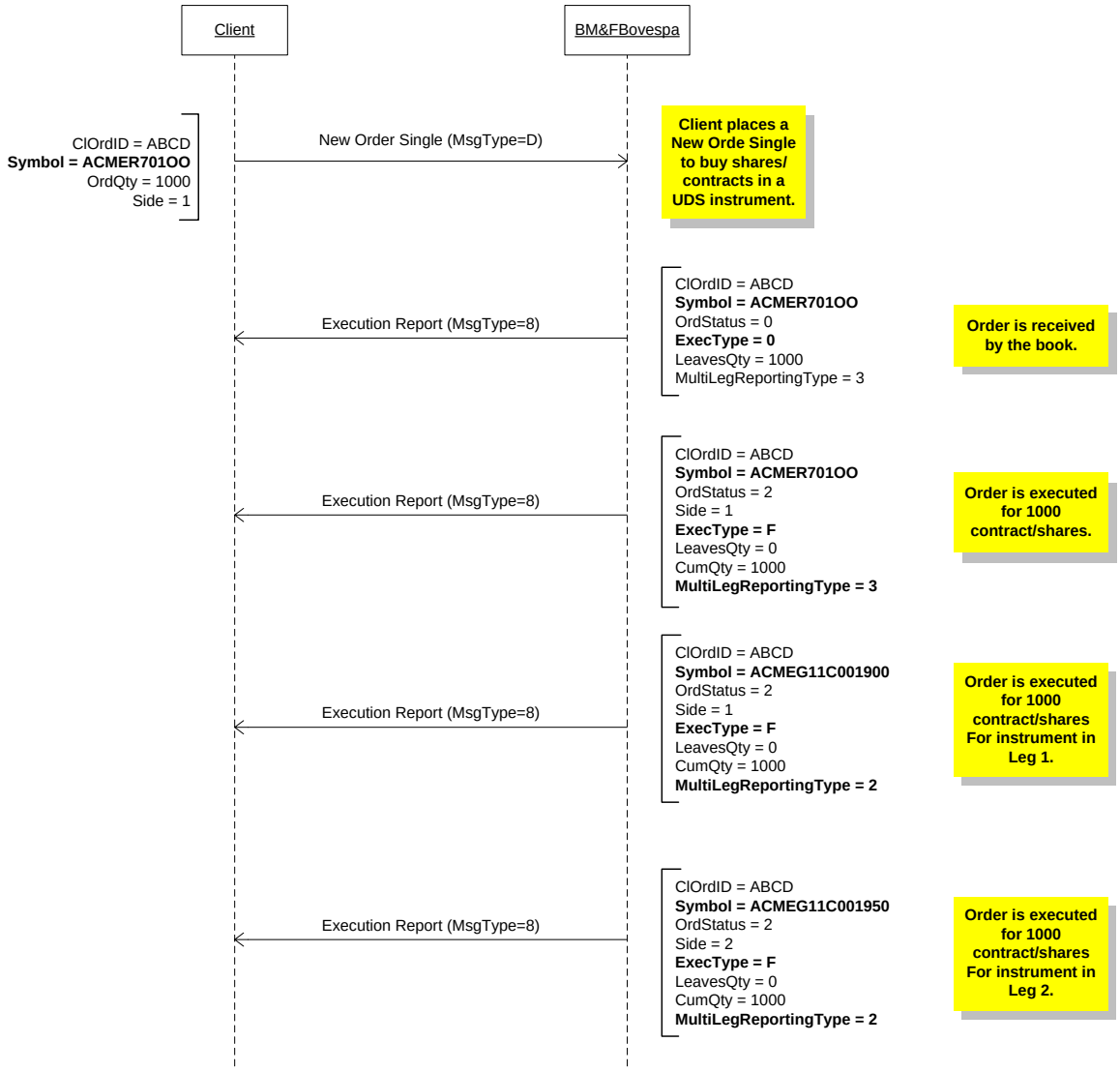

Figure 28 - User-Defined Spread Execution Reports scenario

15.6 Forward

15.6.1 Forward Matching

In the example below, the Initiator sends a Quote Request (35=R) message. BM&FBOVESPA acknowledges the Quote Request with a Quote Status Report message (35=AI) indicating that the Quote Request is in a pending state (297=10). The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote Request ID, Account etc. BM&FBOVESPA then forwards the Quote Request (35=R) to the Respondent's default FIX session based on the Contra Firm ID after removing the Client Account information in Tag 1.

Respondent decides to accept the Declaration and responds back with a Quote message (35=S) with all of the same key details of the deal as contained in the Quote Request (35=R) and with their Client Account. BM&FBOVESPA acknowledges this acceptance with a Quote Status Report message (35=AI) indicating that the Quote (35=S) submitted by the Respondent has been accepted. The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote ID, Account etc. BM&FBOVESPA then forwards the Respondent's Quote message (35=S) to the Initiator's original FIX session after substituting the Client Account information of the Respondent with that of the Initiator so that the Initiator knows that his declaration has been accepted.

Finally, BM&FBOVESPA generates two Execution Reports – one each for the Initiator and Respondent on their respective FIX sessions.

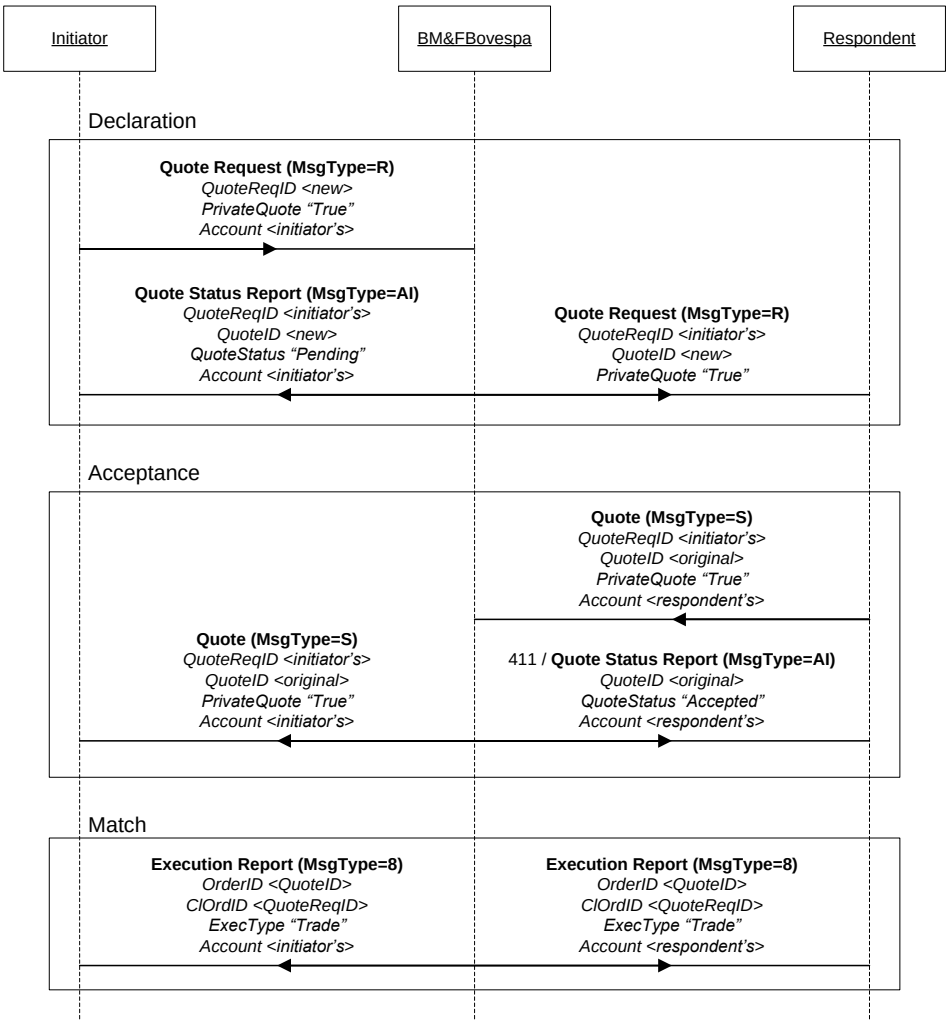

Figure 29 - Forward Matching scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	QuoteID (117)	Side (54)	Account (1)	Firm (452=7)	Contra Firm (452=17)
1	Quote Request (R)		ABC	--	1	12345	100	200
2		QuoteStatus Report (AI)	ABC	DEF	1	12345	100	200
3		QuoteRequest (R)	ABC	DEF	1	--	100	200
4	Quote (S)		ABC	DEF	2	67890	200	100
5		QuoteStatus Report (AI)	ABC	DEF	2	67890	200	100
6		Quote (S)	ABC	DEF	2	12345	200	100
7		ExecutionReport (8)	--	--	1	12345	100	200
8		ExecutionReport (8)	--	--	2	67890	200	100

15.6.2 **Forward Declaration Rejection**

In this example, Initiator sends Declaration as a Quote Request message (35=R) without indicating DaysToSettlement (5497). BM&FBOVESPA rejects this Quote Request with a Quote Request Reject message (35=AG) since DaysToSettlement (5497) is a required field. No Quote ID will be present in this Quote Request Reject message, since the Quote Request itself was never accepted in the first place.

The Quote Request Reject message used to reject Quote Requests submitted by the initiator simply echoes all of the information which was present in the incoming Quote Request such as Quote Request ID, Account, and Order Quantity.

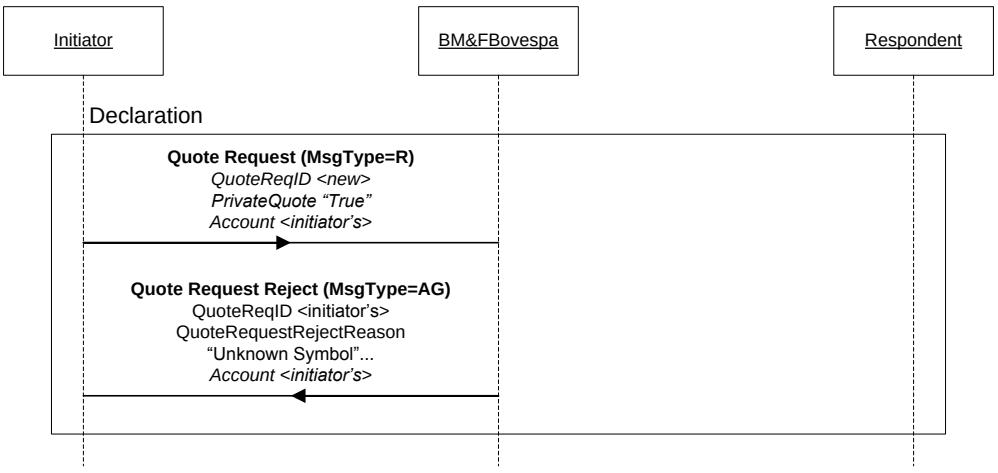

Figure 30 - Forward Declaration Rejection scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	QuoteRequest RejectReason (658)	Comment
1	QuoteRequest (R)		ABC	--	
2		QuoteRequestReject (AG)	ABC	99	Reject

15.6.3 **Forward Acceptance Rejection**

Respondent decides to accept the declaration and responds back with a sell side Quote message (35=S) without confirming the FixedRate (5706). BM&FBOVESPA in turn sends a Quote Status Report message (35=AI) indicating that the Quote (35=S) submitted by the respondent has been rejected since FixedRate is a required tag by using the Quote Request Reject Reason (300) for the error code and Text (58). The Quote Request itself remains in a pending state (297=10) and the Respondent can again attempt to resubmit the Quote. The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote ID, Account etc.

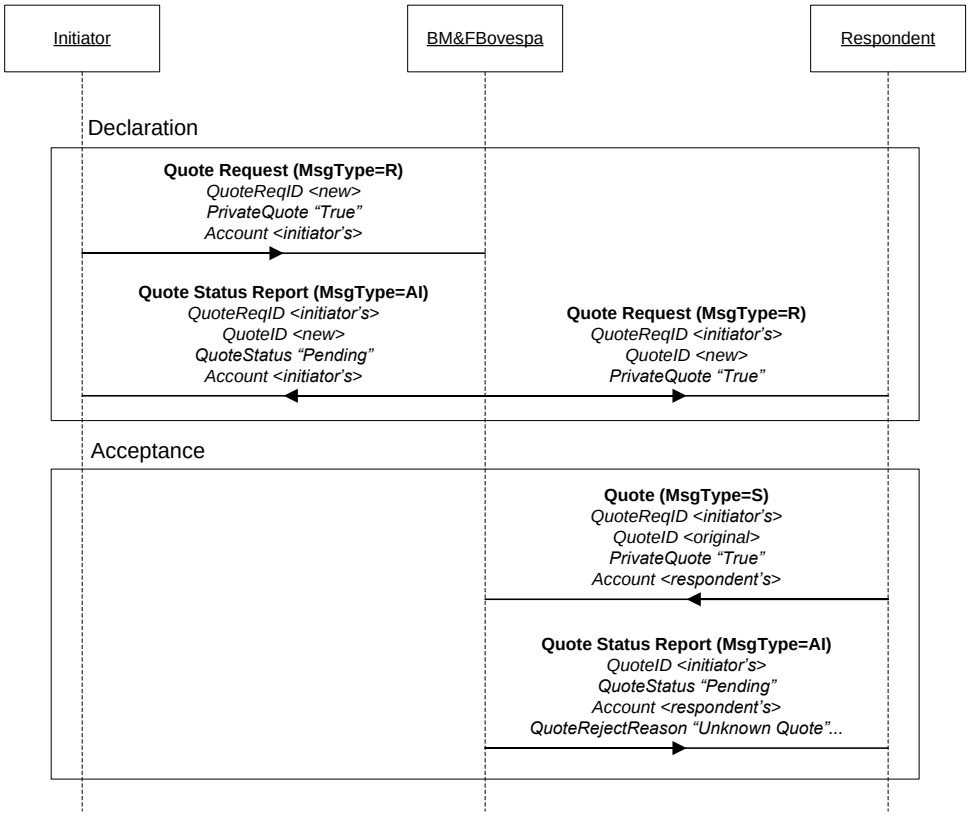

Figure 31 - Forward Acceptance Rejection scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Account (1)	Firm (452=7)	Contra Firm (452=17)	Quote Status (297)	Comment
1	Quote Request (R)		12345	100	200		
2		QuoteStatus Report (AI)	12345	100	200	10	Pending
3		QuoteRequest (R)	--	100	200		
4	Quote (S)		67890	200	100		
5		QuoteStatusReport (AI)	67890	200	100	10	Pending

15.6.4 **Forward Declaration Cancelation**

In order to cancel a Declaration, Initiator sends a Quote Cancel message (35=Z). BM&FBOVESPA cancels the pending Quote Request and sends a Quote Status Report (35=AI) to the Initiator indicating that the Quote Request has been cancelled (297=17). BM&FBOVESPA also forwards the Quote Cancel message (35=Z) to the Respondent's default FIX session based on the Contra Firm ID indicating that the Quote Request has been cancelled and is no longer pending. The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote ID and Account since the Quote Request was found and cancelled. This chain is now finished and the Initiator needs to submit a new Quote Request to begin the dialog again.

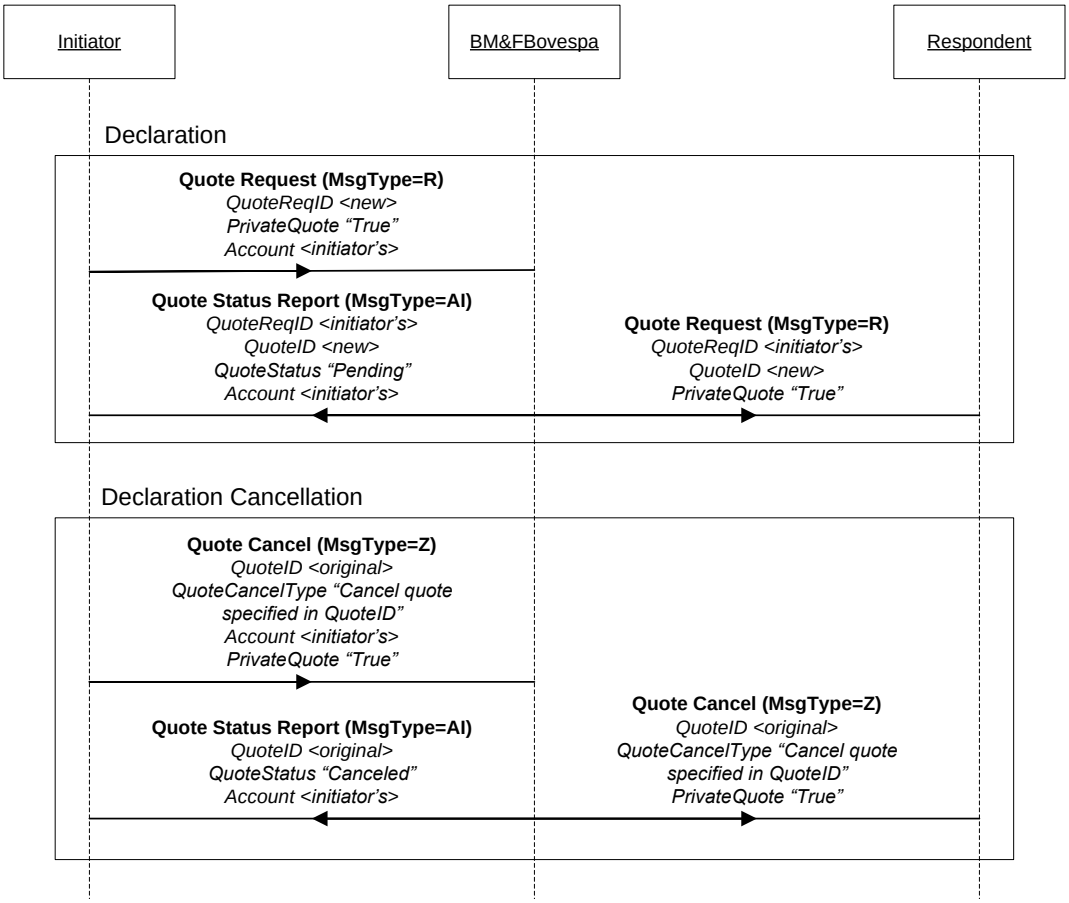

Figure 32 - Forward Declaration Cancelation scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Quote Status (297)	Account (1)	Firm (452=7)	Contra Firm (452=17)	Comment
1	Quote Request (R)		--	12345	100	200	
2		QuoteStatus Report (AI)	10	12345	100	200	Pending
3		QuoteRequest (R)	--	--	100	200	
4	QuoteCancel (Z)		--	12345	100	200	
5		QuoteStatus Report (AI)	17	12345	100	200	Cancelled
6		QuoteCancel (Z)	--	--	100	200	

15.6.5 **Forward Declaration Cancelation Rejection**

Initiator sends a Quote Cancel message (35=Z) to cancel the pending Quote Request. BM&FBOVESPA rejects this Quote Cancel message (35=Z) since it contains the wrong Quote ID, for example, and sends a Quote Status Report (35=AI) to the Initiator indicating that the Quote Request is still pending (297=10). The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote ID and Account. The Initiator can again attempt to resubmit a well formed Quote Cancel message to cancel the pending Quote Request.

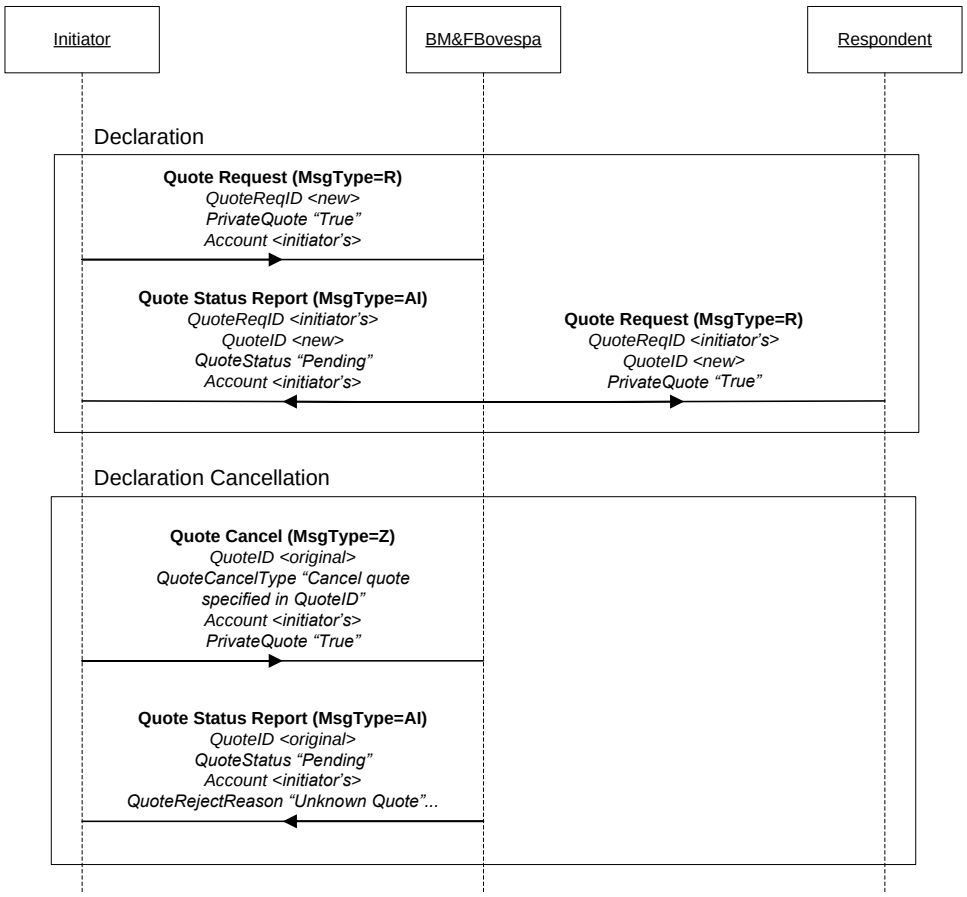

Figure 33 - Forward Declaration Cancelation Rejection scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	QuoteID (117)	Quote Status (297)	Account (1)	Comment
1	Quote Request (R)		ABC		--	12345	
2		QuoteStatus Report (AI)	ABC	DEF	10	12345	Pending
3		Quote Request (R)	ABC	DEF	--	--	
4	Quote Cancel (Z)		--	XYZ	--	12345	
5		QuoteStatus Report (AI)	NONE	XYZ	9	12345	Quote Not Found

15.6.6 **Forward Counterparty Refusal**

In this example, the Respondent does not agree with the proposed terms of the deal and decides to reject the declaration. The Respondent replies with a Quote Request Reject message (35=AG) which indicates that he has decided to Pass (658=10) on this particular Quote Request (35=R). BM&FBOVESPA then acknowledges this by sending a Quote Status Report message (35=AI) back to the Respondent confirming that the Quote Request (35=R) has been passed upon (297=11). BM&FBOVESPA also forwards the Quote Request Reject message (35=AG) back to the Initiator's original FIX session to inform them that their Quote Request (35=R) was passed upon (297=11) by the Respondent. This chain is now finished and the Initiator needs to submit a new Quote Request to begin the dialog again.

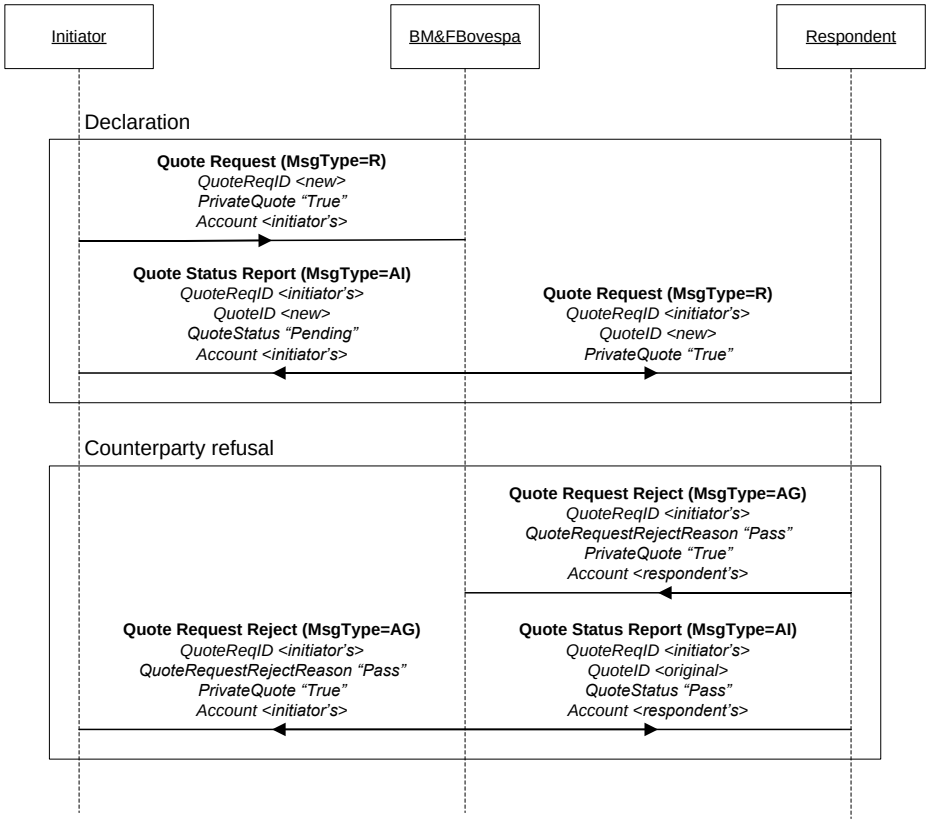

Figure 34 - Forward Counterparty Refusal scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Quote Status (297)	Account (1)	Firm (452=7)	Contra Firm (452=17)	Comment
1	QuoteRequest (R)		--	12345	100	200	
2		QuoteStatusReport (AI)	10	12345	100	200	Pending
3		QuoteRequest (R)	--	--	100	200	
4	QuoteRequest Reject (AG)		--	67890	200	100	
5		QuoteStatusReport (AI)	11	67890	200	100	Pass
6		QuoteRequestReject (AG)	--	12345	200	100	

15.6.7 **Forward Counterparty Refusal Reject**

In this scenario, the Quote Request Reject message sent by the Respondent gets rejected because it has been sent with an invalid QuoteReqID, for example. BM&FBOVESPA then sends a Quote Status Report message (35=AI) back to the Respondent indicating that the Quote Request (35=R) could not be found (297=9). The Quote Status Report message sent to Respondent simply echoes all of the information which was present in the corresponding incoming message. The Quote Reject Reason (300) and Text (58) fields in the Quote Status Report message are used to convey the rejection details. The Respondent can again attempt to resubmit a well formed Quote Request Reject message to pass on the pending Quote Request.

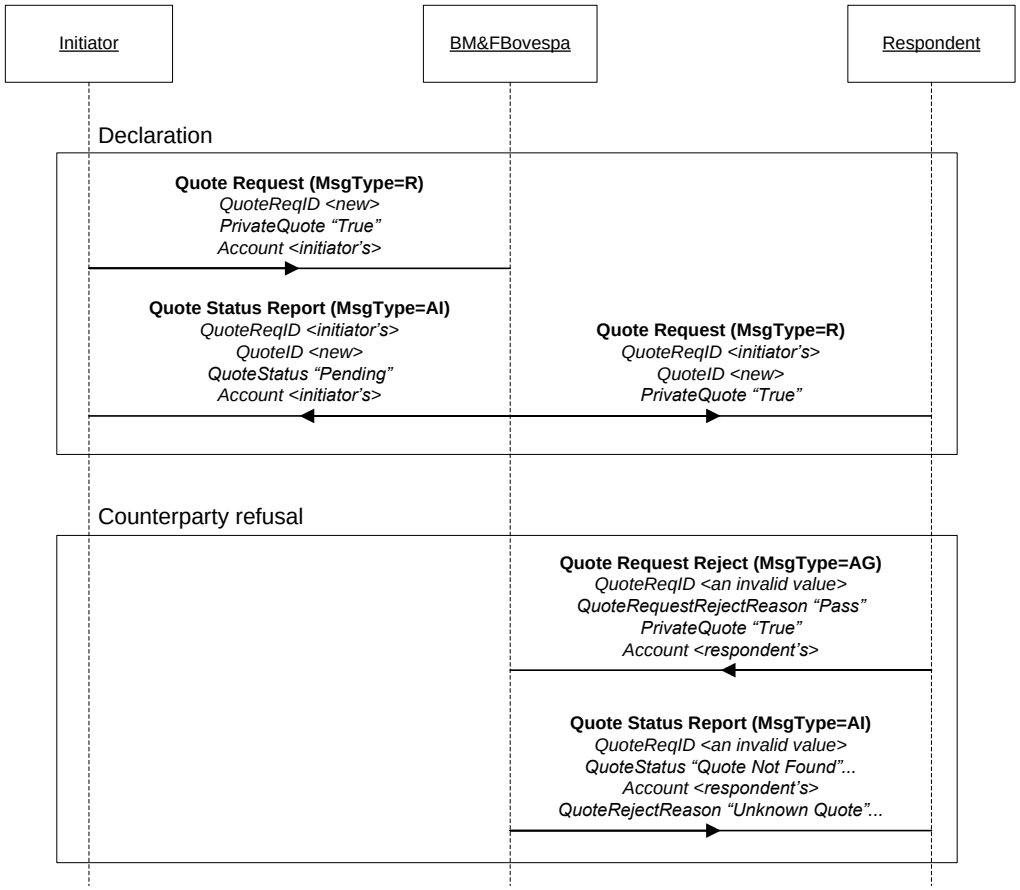

Figure 35 - Forward Counterparty Refusal Rejection scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Quote Status (297)	Quote RejectReason (300)	Account (1)	Comment
1	QuoteRequest (R)		--	--	12345	
2		QuoteStatusReport (AI)	10	--	12345	Pending
3		QuoteRequest (R)	--	--	--	
4	QuoteRequest Reject (AG)		--	--	67890	
5		QuoteStatusReport (AI)	9	1	67890	Quote not Found

15.6.8 **Forward Expiration**

In case the Respondent does not reply back to the Quote Request (35=R) submitted by the Initiator until the end of the trading session, the pending Quote Request (35=R) will be eliminated and BM&FBOVESPA will send a Quote Status Report message (35=AI) to both parties, Initiator and Respondent, informing them that the Quote Request has expired (297=7). This chain is now finished and the Initiator needs to submit a new Quote Request to begin the dialog again.

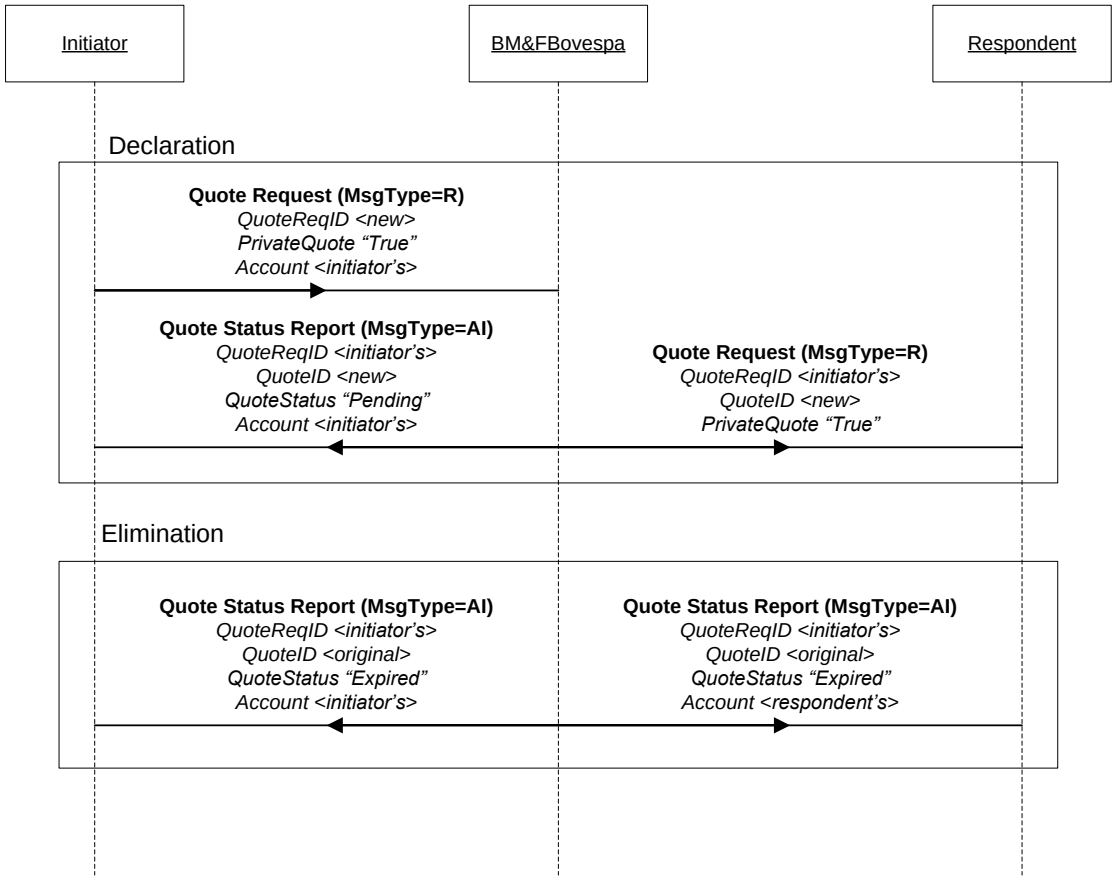

Figure 36 - Forward Expiration scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	QuoteID (117)	QuoteStatus (297)	Account (1)	Comment
1	QuoteRequest (R)		ABC	--	--	12345	
2		QuoteStatus Report (AI)	ABC	DEF	10	12345	Pending
3		QuoteRequest (R)	ABC	DEF	--	--	
4		QuoteStatus Report (AI)	ABC	DEF	7	12345	Expired
5		QuoteStatus Report (AI)	ABC	DEF	7	--	Expired

15.6.9 **Forward Trade Bust**

This example illustrates the system behavior when trades are cancelled by Market Ops after a successful Forward deal has been completed. In this case, trade cancel reports are sent by BM&FBOVESPA to the Initiator and Respondent on their original FIX sessions respectively.

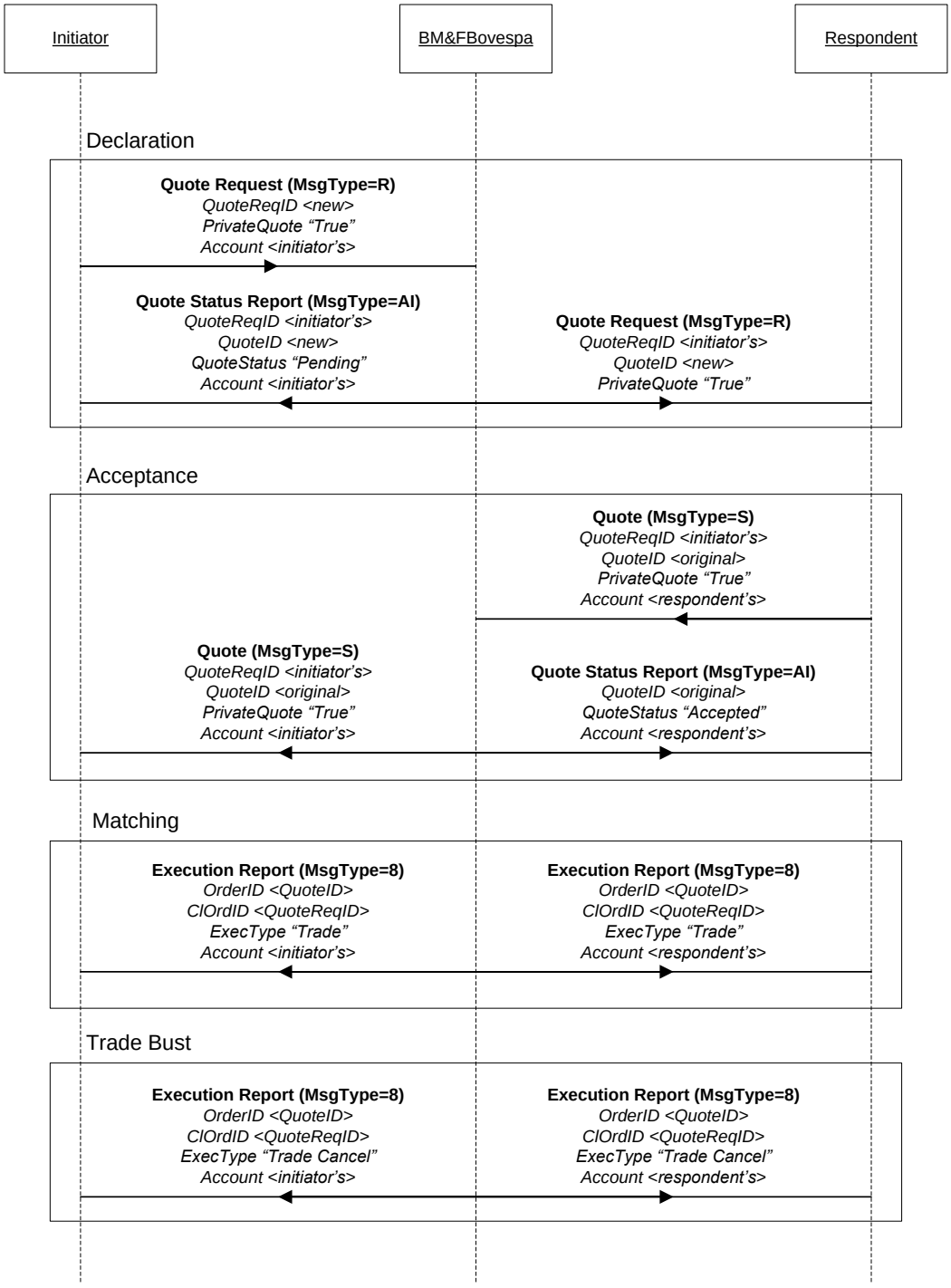

Figure 37 - Forward Trade Bust scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	QuoteID (117)	Account (1)	Comment
1	QuoteRequest (R)		ABC	--	12345	
2		QuoteStatusReport (AI)	ABC	DEF	12345	Pending
3		QuoteRequest (R)	ABC	DEF	--	
4	Quote (S)		ABC	DEF	67890	
5		QuoteStatusReport (AI)	ABC	DEF	67890	Accept
6		Quote (S)	ABC	DEF	12345	
7		ExecutionReport (8)	--	--	12345	Fill
8		ExecutionReport (8)	--	--	67890	Fill
9		ExecutionReport (8)	--	--	12345	Trade Cancel
10		ExecutionReport (8)	--	--	67890	Trade Cancel

15.6.10 Cross Forward

Initiator sends a two sided cross declaration as a Quote Request message (35=R). BM&FBOVESPA acknowledges the Quote Request with two Quote Status Report messages (35=AI) – one for each side – indicating that the Quote Request has been accepted (297=0). Then BM&FBOVESPA publishes two trade execution reports to the Initiator's original FIX session – one for each side.

The second side of a Cross Forward Quote Request only needs to contain minimal information – i.e. only fields which are different from the first side, such as Account, for example, need to be repeated in the second side, since it will be assumed that the absent fields in the second repeating group default to the same values as mentioned in the first repeating group.

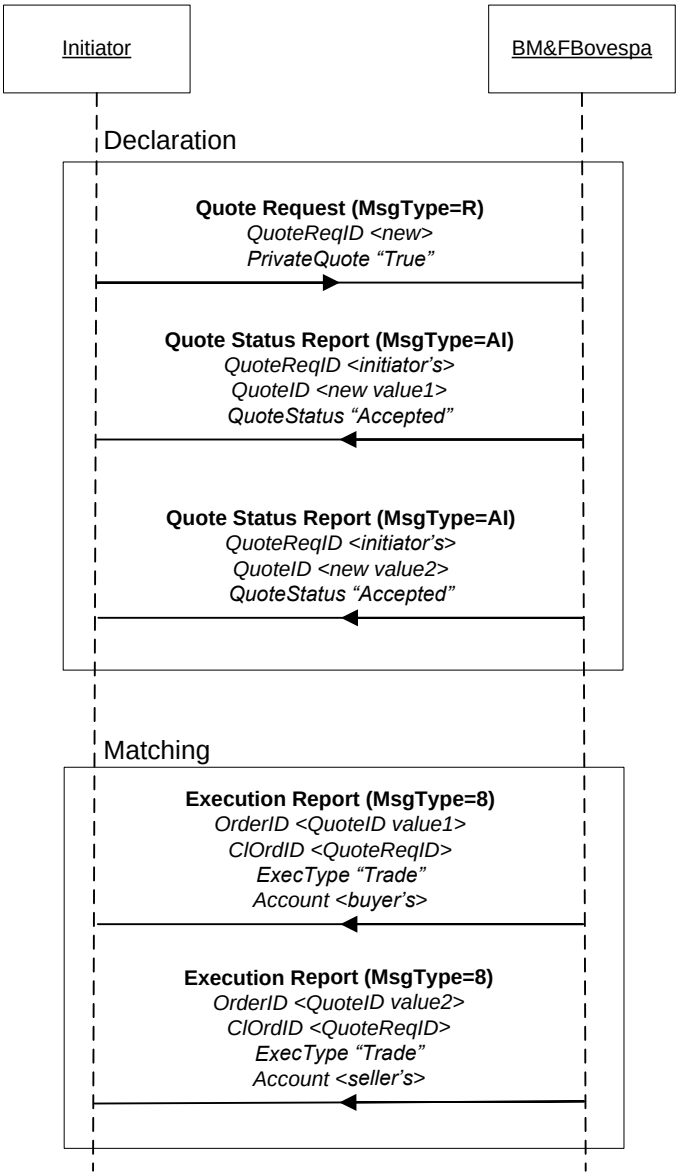

Figure 38 - Cross Forward scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Side (54)	Account (1)	Firm (452=7)	Contra Firm (452=17)	Comment
1	QuoteRequest (R)		1	12345	100	100	
			2	67890			
2		QuoteStatusReport (AI)	1	12345	100	100	Accept
3		QuoteStatusReport (AI)	2	67890	100	100	Accept
4		ExecutionReport (8)	1	12345	100	100	Fill
5		ExecutionReport (8)	2	67890	100	100	Fill

15.6.11 Cross Forward Rejection

In the following diagram, the Initiator sends a two sided Cross declaration as a Quote Request message (35=R) without indicating the Settlement Type (Tag 63) in the first repeating group, for example. BM&FBOVESPA then rejects this Quote Request with a single Quote Request Reject message (35=AG) since Settlement Type (Tag 63) is a required tag. The Quote Request Reject message simply echoes all information which was present in the corresponding incoming message. The Quote Request Reject Reason (658) and Text (58) fields in the Quote Request Reject message are used to convey the rejection details.

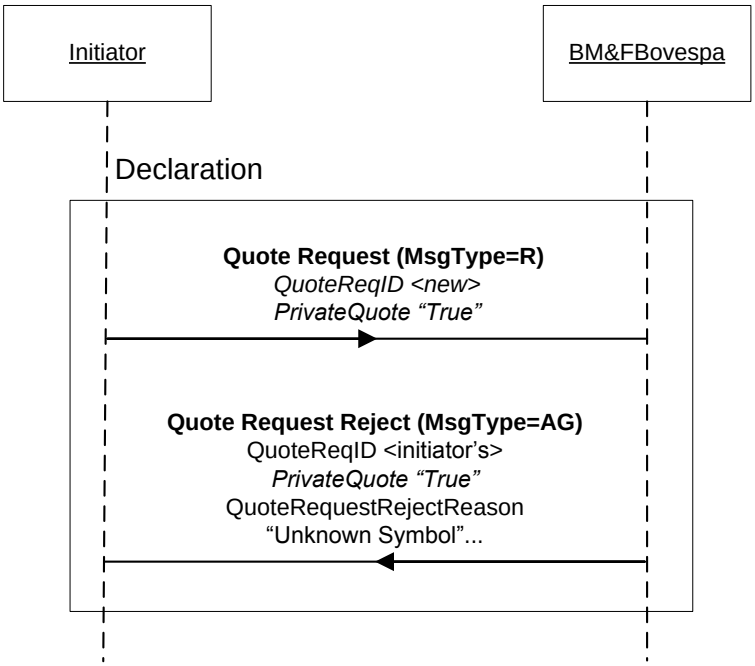

Figure 39 - Cross Forward Rejection scenario

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	QuoteReqID (131)	Side (54)	Account (1)	Firm (452=7)	Contra Firm (452=17)	Comment
1	QuoteRequest (R)		ABC	1	12345	100	200	
				2	67890			
2		QuoteRequest Reject (AG)	ABC	1	12345	100	200	Reject
				2	67890			

15.6.12 Forward + Cash ("Termo Vista")

In the example below, the Initiator sends a Quote Request (35=R) message with tag ExecuteUnderlyingTrade (35004) as an indicative that it's a Forward + Cash operation. BM&FBOVESPA acknowledges the Quote Request with a Quote Status Report message (35=AI) indicating that the Quote Request is in a pending state (297=10). The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote Request ID, Account etc. BM&FBOVESPA then forwards the Quote Request (35=R) to the Respondent's default FIX session based on the Contra Firm ID after removing the Client Account information in Tag 1.

Respondent decides to accept the Declaration and responds back with a Quote message (35=S) with all of the same key details of the deal as contained in the Quote Request (35=R) and with their Client Account. BM&FBOVESPA acknowledges this acceptance with a Quote Status Report message (35=AI) indicating that the Quote (35=S) submitted by the Respondent has been accepted. The outgoing Quote Status Report message simply echoes all of the information which was present in the corresponding incoming message such as Quote ID, Account etc. BM&FBOVESPA then forwards the Respondent's Quote message (35=S) to the Initiator's original FIX session after substituting the Client Account information of the Respondent with that of the Initiator so that the Initiator knows that his declaration has been accepted.

Then BM&FBOVESPA generates two Execution Reports – one each for the initiator and respondent on their respective FIX sessions. BM&FBOVESPA publishes two trade Execution Reports for the underlying cash instrument also – one each for the Initiator and Respondent on their original FIX sessions respectively, inverting the buyer and seller, i.e., the Forward buyer becomes the cash seller, and the Forward seller becomes the cash buyer. The trade Execution Reports for Forward and Cash contain the same identifiers such as ClOrdID will contain the value of QuoteReqID and OrderID will have the same value of QuoteID.

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	Symbol (55)	Side (54)	Account (1)	Firm (452=7)	Contra Firm (452=17)	Comment
1	Quote Request (R)		ACME4T	1	12345	100	200	
2		QuoteStatus Report (AI)	ACME4T	1	12345	100	200	Pending
3		QuoteRequest (R)	ACME4T	1	--	100	200	
4	Quote (S)		ACME4T	2	67890	200	100	
5		QuoteStatus Report (AI)	ACME4T	2	67890	200	100	Accept
6		Quote (S)	ACME4T	2	12345	200	100	
7		ExecutionReport (8)	ACME4T	1	12345	100	200	Forward Fill
8		ExecutionReport (8)	ACME4T	2	67890	200	100	Forward Fill
9		ExecutionReport (8)	ACME4	2	12345	100	200	Cash Fill
10		ExecutionReport (8)	ACME4	1	67890	200	100	Cash Fill

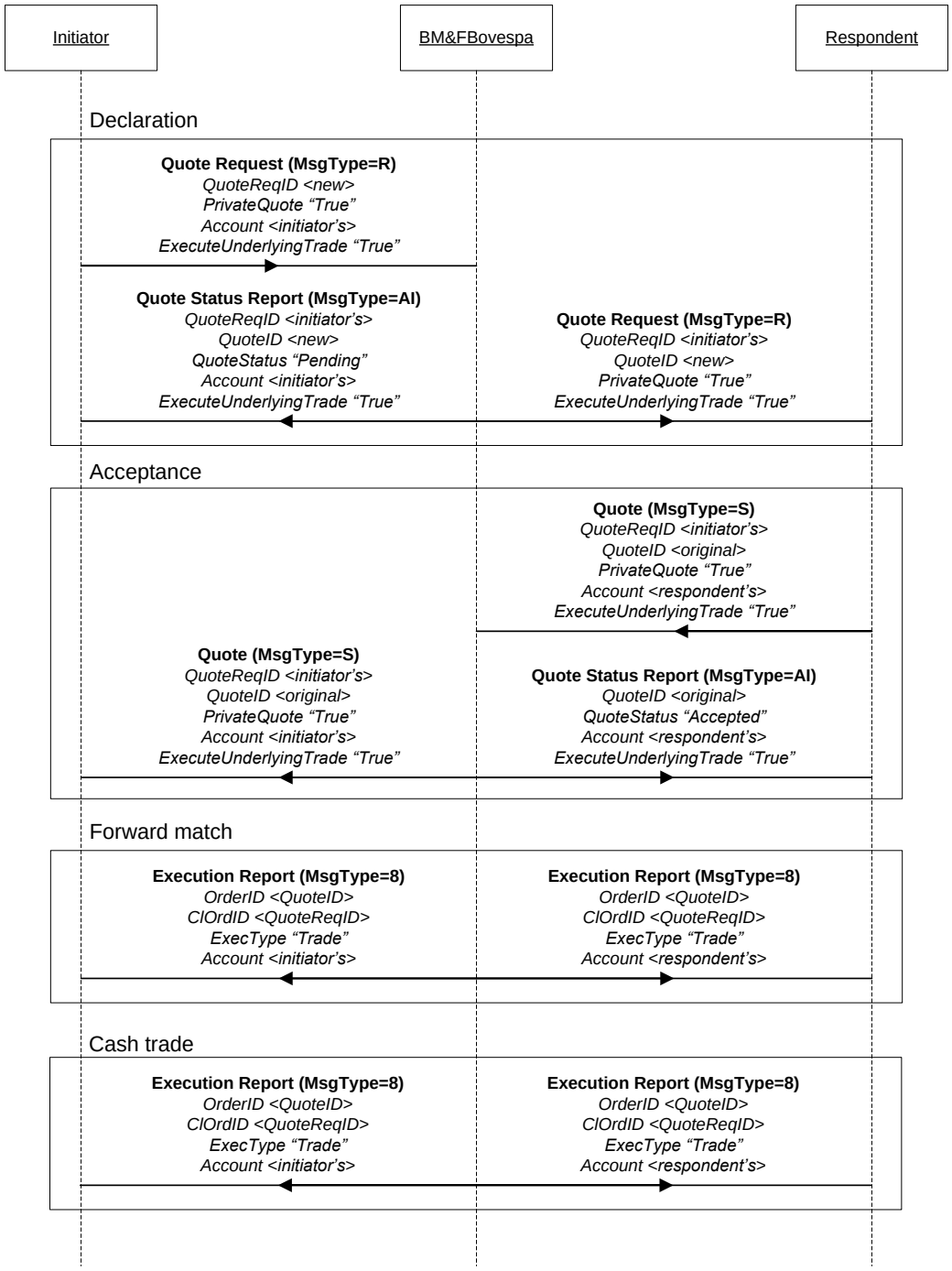

Figure 40 - Forward + Cash scenario

15.6.13 Forward + Registered Cash ("Termo Vista Registered")

In the following example, Initiator has bought already a security in the cash market and wants to trade using "Forward + Registered Cash" (TVR).

Initiator sends a buy side declaration as a Quote Request message (35=R) indicating the number of the cash trade in tag Unique Trade ID (6032) and setting tag Execute Underlying Trade (35004) to 1. BM&FBOVESPA acknowledges the Quote Request with a Quote Status Report message (35=AI) indicating that the Quote Request is in a pending state (297=10). BM&FBOVESPA will then forward the Quote Request (35=R) to the respondent's default FIX session based on the Contra Firm ID after removing the Client Account information in Tag 1 and the Unique Trade ID in tag 6032.

Respondent decides to accept the declaration and responds back with a sell side Quote message (35=S) with all of the same key details of the deal as contained in the Quote Request (35=R) and with their Client Account. BM&FBOVESPA acknowledges this acceptance with a Quote Status Report message (35=AI) indicating that the Quote (35=S) submitted by the respondent has been accepted. BM&FBOVESPA then forwards the respondent's Quote message (35=S) back to the initiator's original FIX session after substituting the Client Account information of the respondent with that of the initiator so that the initiator knows that his declaration has been accepted.

After this BM&FBOVESPA publishes two execution reports for the Termo instrument itself – one each for the initiator (buy side) and respondent (sell side) on their original FIX sessions respectively. Then BM&FBOVESPA publishes two trade execution reports for the underlying cash instrument also – one each for the initiator (sell side) and respondent (buy side) on their original FIX sessions respectively since the trade for Termo Vista inverts the buyer and seller, i.e., the forward buyer becomes the cash seller, and the forward seller becomes the cash buyer.

The trade execution reports for Termo and Vista will contain the same identifiers such as Client Order ID = Quote Request ID and Order ID = Quote ID.

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

Time	Message Received	Message Sent	55	54	1	35004	6032	448 (452=7)	448 (452=17)	Comment
1	Quote Request (R)		ACME 4T	1	12345	1	70	100	200	
2		QuoteStatus Report (AI)	ACME 4T	1	12345	1	70	100	200	Pending
3		Quote Request (R)	ACME 4T	1	--	1	--	100	200	
4	Quote (S)		ACME 4T	2	67890	1	--	200	100	
5		QuoteStatus Report (AI)	ACME 4T	2	67890	1	--	200	100	Accept
6		Quote (S)	ACME 4T	2	12345	1	--	200	100	
7		Execution Report (8)	ACME 4T	1	12345	--	--	100	200	Termo Fill
8		Execution Report (8)	ACME 4T	2	67890	--	--	200	100	Termo Fill
9		Execution Report (8)	ACME 4	2	12345	--	--	100	200	Termo Vista Fill
10		Execution Report (8)	ACME 4	1	67890	--	--	200	100	Termo Vista Fill

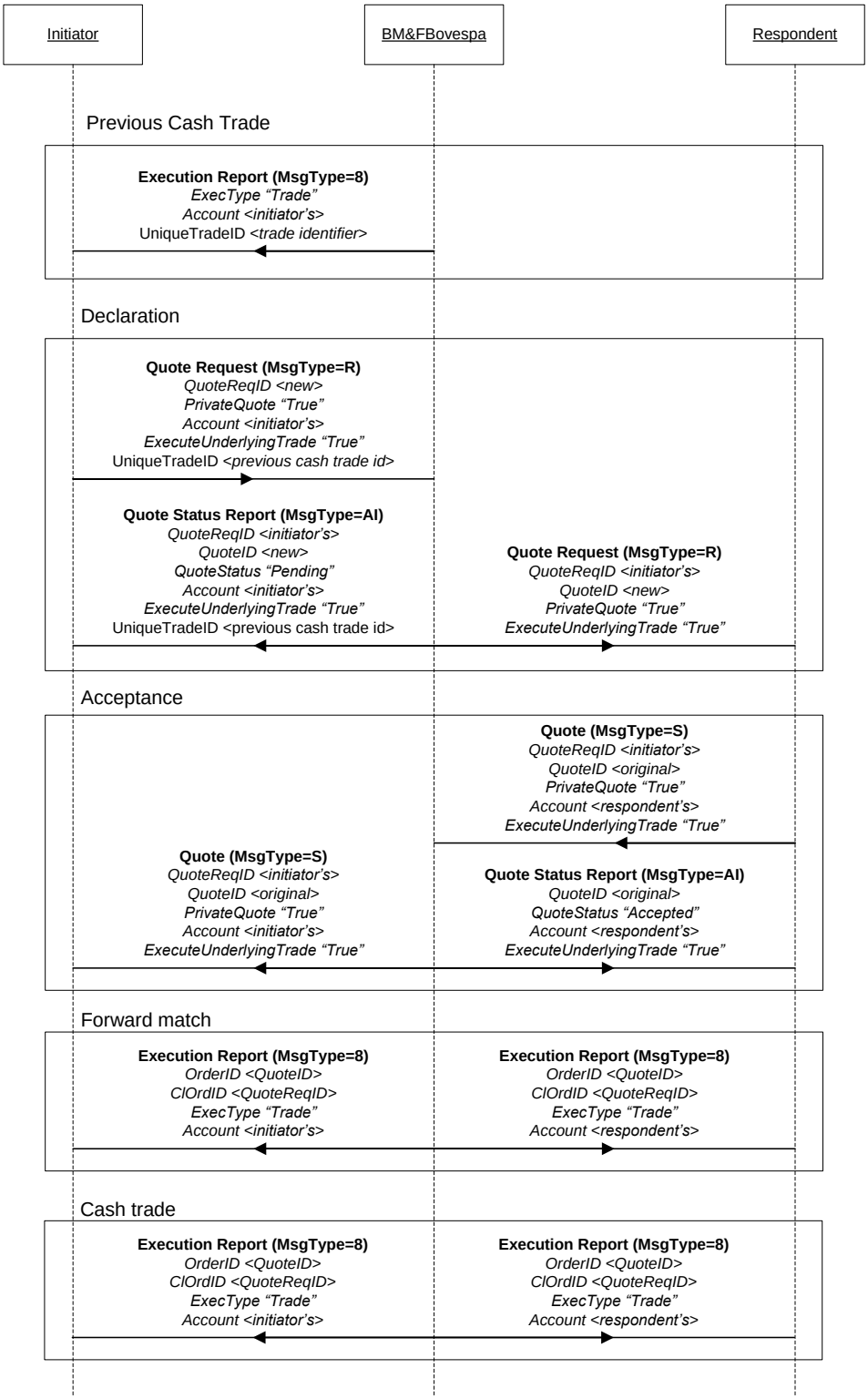

Figure 41 - Forward + Registered Cash scenario

15.7 Self-Trading prevention

This section presents scenarios where the Self-Trading prevention at the customer level causes the cancelation of orders entered by customers. In all cases, the system sends an ExecutionReport and provides the reason for the order elimination in tag ExecRestatementReason (378).

15.7.1 Self-Trading prevention on Aggressing Order

In this scenario, the customer already has an order in the book tagged with their unique Investor ID and the new order that is being sent can potentially match with the one in the book.

In this case, the new order is accepted and then cancelled upon entry. Tag OrderStatus (39) in the ExecutionReport sent to the participant indicates that the order has been cancelled (39 = 4 Cancelled) and tag ExecRestatementReason (378) provides a self-explanatory reason for the elimination (378 = 103 Self-Trading prevention).

Note that the system does not run Self-Trading validations at customer level for orders not tagged with Party Role (452) = 5 Investor ID.

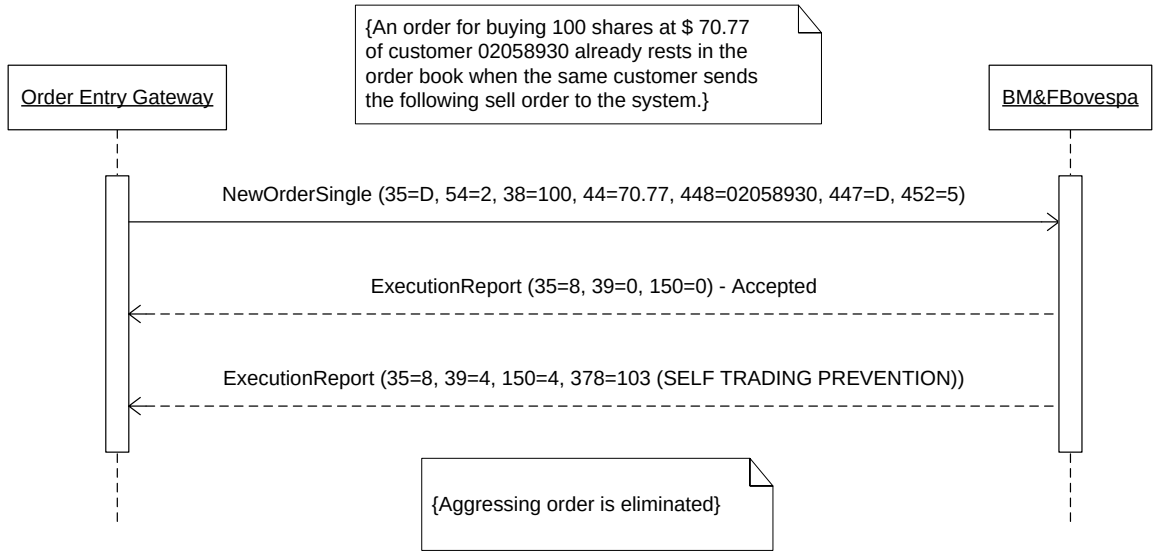

Figure 42 - Self-Trading prevention on aggressing order

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	40	448 (452=5)	38	44	39	150	378	Comment
1	NewOrderSingle (D)		2	02058930	100	70.77	--	--	--	
2		ExecutionReport (8)	2	02058930	100	70.77	0	0	--	New
3		ExecutionReport (8)	2	02058930	100	70.77	4	4	103	Cancelled

15.7.2 **Self-Trading prevention on Order Modification**

When an order modification leads to a potential match with another order, tagged with the same Investor ID, the modification will be accepted but it will be followed by an immediate elimination of the order.

Tag OrderStatus (39) in the ExecutionReport sent to the participant indicates that the order has been cancelled (39 = 4 Cancelled) and tag ExecRestatementReason (378) provides a self-explanatory reason for the elimination (378 = 103 Self-Trading prevention).

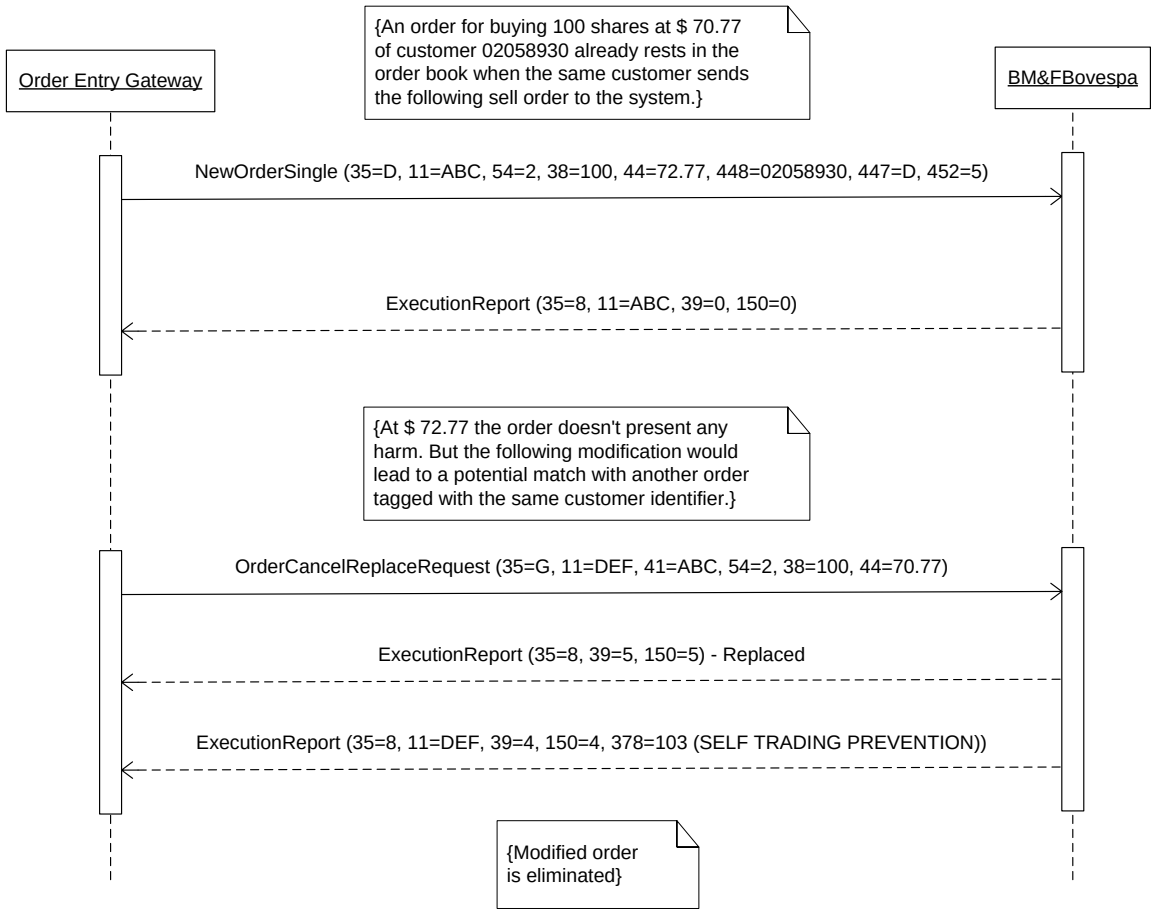

Figure 43 - Self-Trading prevention on order modification

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	11	40	448 (452=5)	38	44	39	150	378	Comment
1	NewOrderSingle (D)		ABC	2	02058930	100	72.77	--	--	--	
2		ExecutionReport (8)	ABC	2	02058930	100	72.77	0	0	--	New
3	OrderCancelReplace Request (G)		DEF	2	02058930	100	70.77	--	--	--	
4		ExecutionReport (8)	DEF	2	02058930	100	70.77	5	5	--	Replaced
5		ExecutionReport (8)	DEF	2	02058930	100	70.77	4	4	103	Cancelled

15.7.3 **Self-Trading prevention and Partial Fills**

This scenario presents a situation in which the order is partially executed in 200 shares and the remaining amount of 800 is eliminated because the next aggressed order has the same unique Investor ID.

Tag OrderStatus (39) in the ExecutionReport sent to the participant indicates that the order has been cancelled (39 = 4 Cancelled) and tag ExecRestatementReason (378) provides a self-explanatory reason for the elimination (378 = 103 Self-Trading prevention).

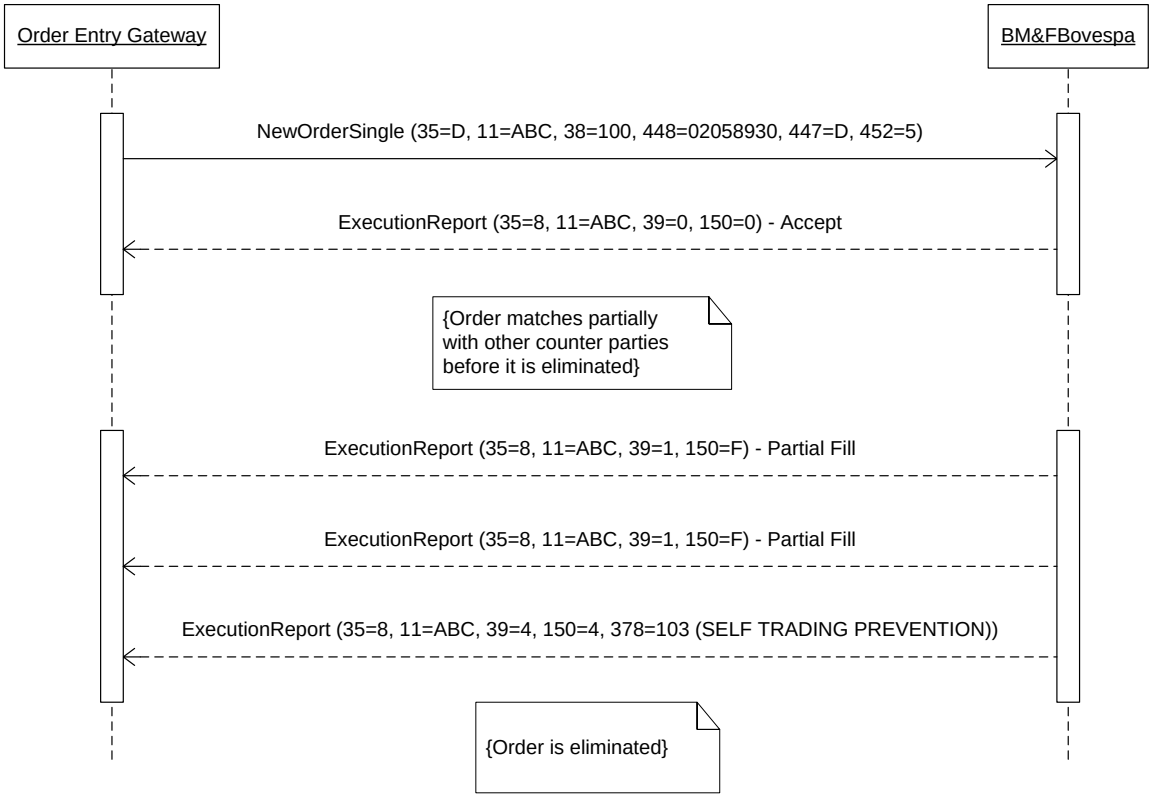

Figure 44 - Self-Trading prevention and partial fills

For incoming Minimum quantity orders or FOK orders, the Match Engine analyzes the book to assure the minimum required quantities can be achieved without self-trade. Otherwise the incoming (aggressing) order must be eliminated upon entry.

The following table shows the sequence of messages received and sent by the Exchange and some sample values are assigned to key fields in order to demonstrate their usage:

	Message Received	Message Sent	11	448 (452=5)	38	32	151	14	39	150	378	Comment
1	NewOrderSingle (D)		ABC	02058930	1000	--	--	--	--	--	--	
2		ExecutionReport (8)	ABC	02058930	1000	--	1000	--	0	0	--	New
3		ExecutionReport (8)	ABC	02058930	1000	100	900	100	1	F	--	Partial Fill
4		ExecutionReport (8)	ABC	02058930	1000	100	800	200	1	F	--	Partial Fill
5		ExecutionReport (8)	ABC	02058930	1000	--	800	200	4	4	103	Cancelled

15.7.4 **Self-Trading prevention on Stop Orders**

In the following scenario, a Stop order becomes a Limit order and it is tagged with the same Investor ID as another order in the book. If such condition leads to a self-trade, the triggered order will be immediately eliminated.

Tag WorkingIndicator (636) points when the order becomes active and available for trading. Upon activation, the OrderType (40) changes from “Stop Limit” (40 = 4) to “Limit” (40 = 2) and the order is eliminated by the Self-Trading prevention functionality.

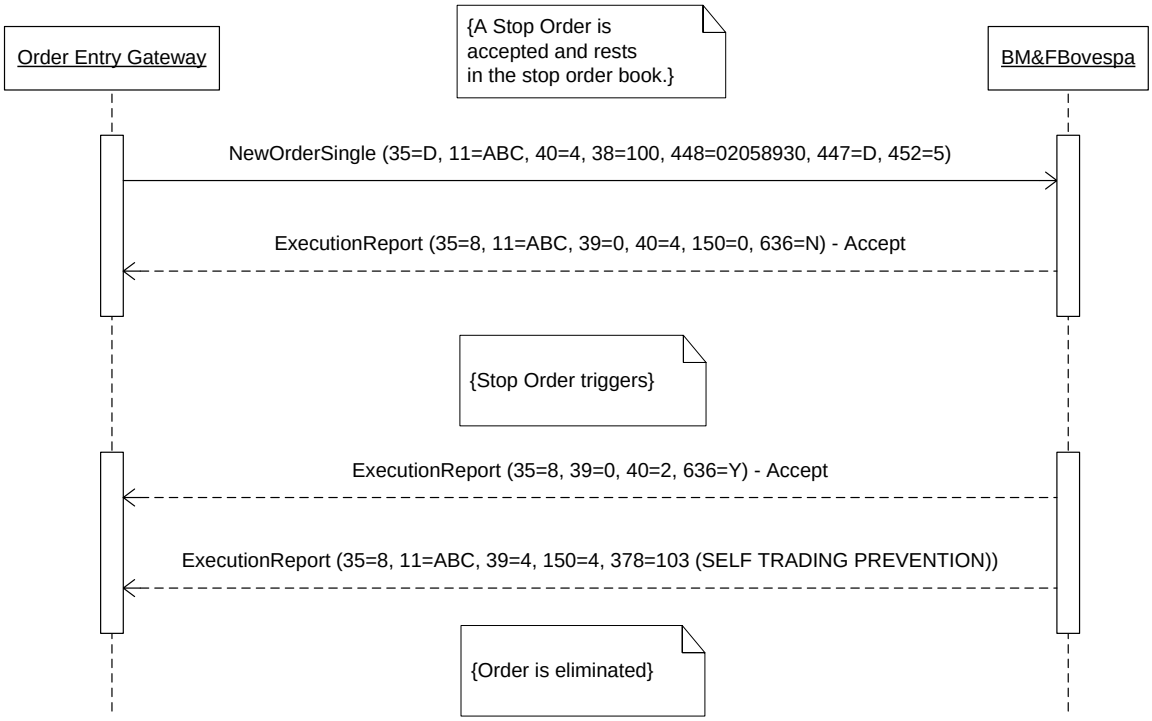

Figure 45 - Self-Trading prevention on Stop orders

	Message Received	Message Sent	11	40	448 (452=5)	38	39	150	378	636	Comment
1	NewOrderSingle (D)		ABC	4	02058930	100	--	--	--	--	
2		ExecutionReport (8)	ABC	4	02058930	100	0	0	--	N	New
3		ExecutionReport (8)	ABC	2	02058930	100	0	0	--	Y	New
4		ExecutionReport (8)	ABC	2	02058930	100	4	4	103	--	Cancelled

15.8 Message Replay

15.8.1 Retransmission Request

In this example, client on a FIX session named C555AAA requests for message re-send from the FIX session C999AAC.

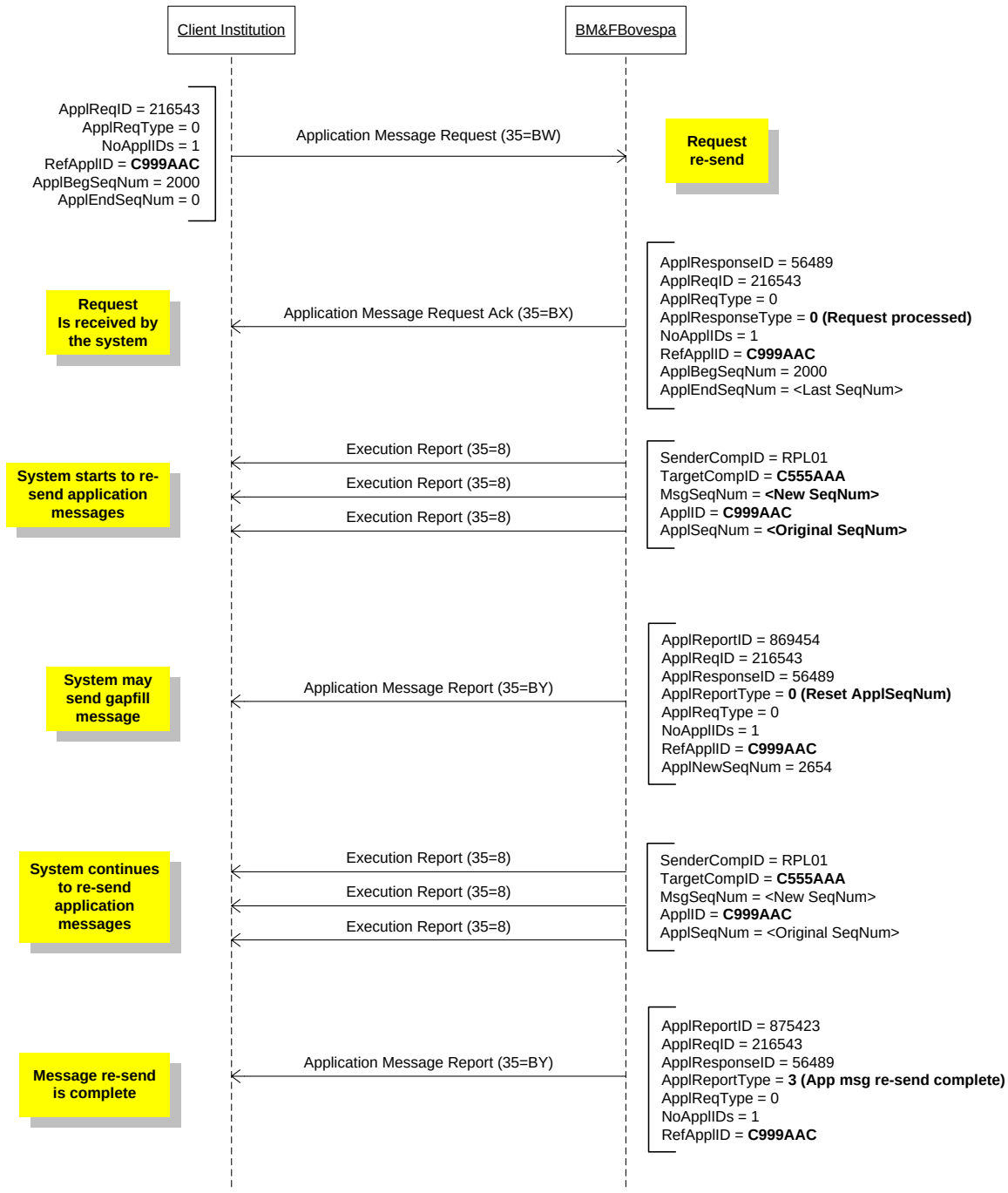

Figure 46 - Message Re-send

15.8.2 Rejection Scenarios

The Application Message Request Acknowledgment (tag 35=BX) message is sent to confirm the receiving of the Application Message Request (tag 35=BW) message.

The requested messages are resent only when the value of ApplResponseStatus (tag 35021) is “0” (Request Successfully Processed).

The other values for tag 35021 indicate negative acknowledgment.

15.8.2.1 User Not Authorized

In the example below, the Application Message Request is rejected because the user is not authorized to request re-send from a particular FIX session.

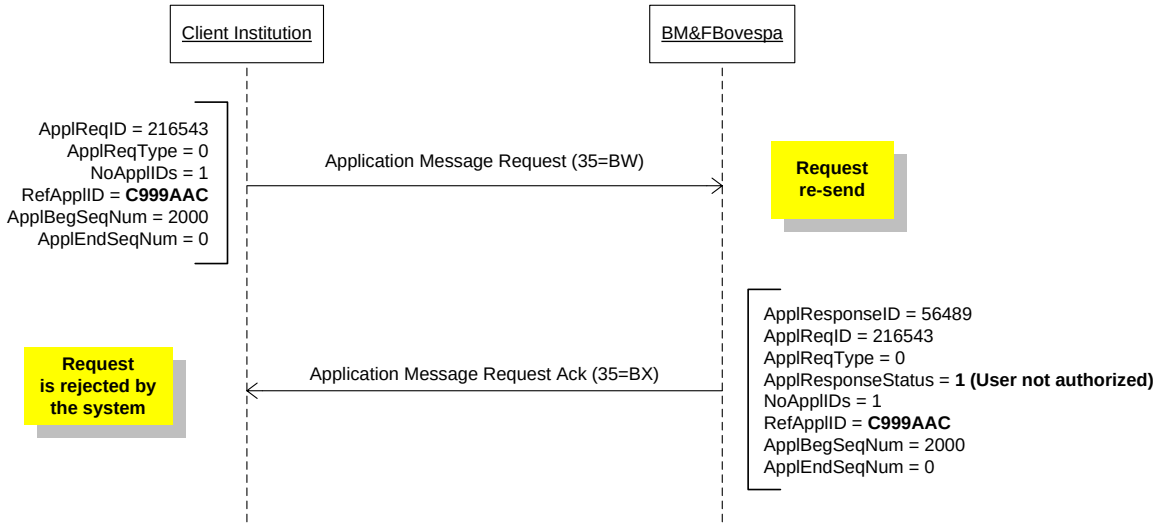

Figure 47 - User not authorized

15.8.2.2 Invalid Range Requested

The following picture presents an Application Message Request being rejected due to an invalid parameter defined for the range of application sequence numbers.

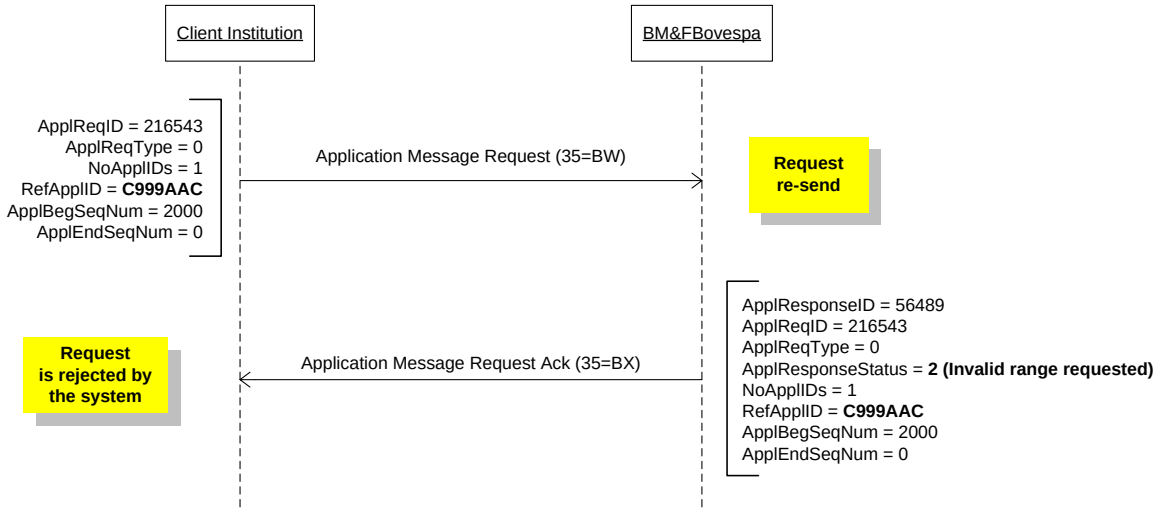

Figure 48 - Invalid range requested

15.8.2.3 Re-send already in progress

The system will reject any request for retransmission of messages that are currently in progress.

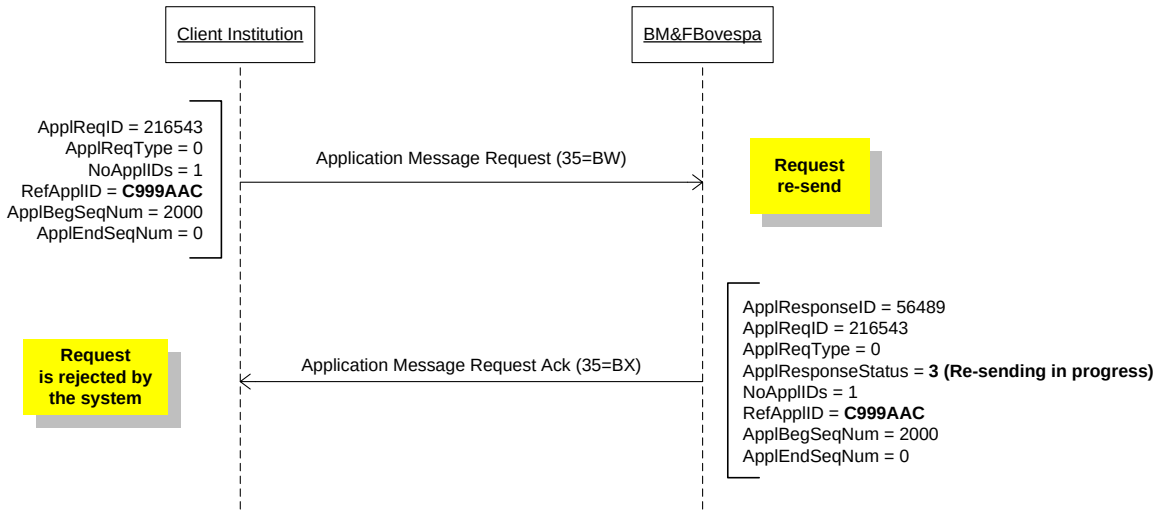

Figure 49 - Re-send already in progress

15.8.3 Error Scenarios

The ApplReportType field (tag 1426) in Application Message Report (tag 35=BY) reports whether the resending was successfully completed (value=3) or there was an error (value=4).

This message might be sent immediately after the Application Message Request Acknowledgment (tag 35=BX) message (if an error occurs and messages cannot be resent), or in the middle of a transmission that was interrupted because of an error.

15.8.3.1 Error Before Re-sending

If after sending Application Message Request Ack the system detects an error that hampers the message retransmission, an Application Message Report will be sent to the client with tag ApplReportType = 4 and no messages will be resent.

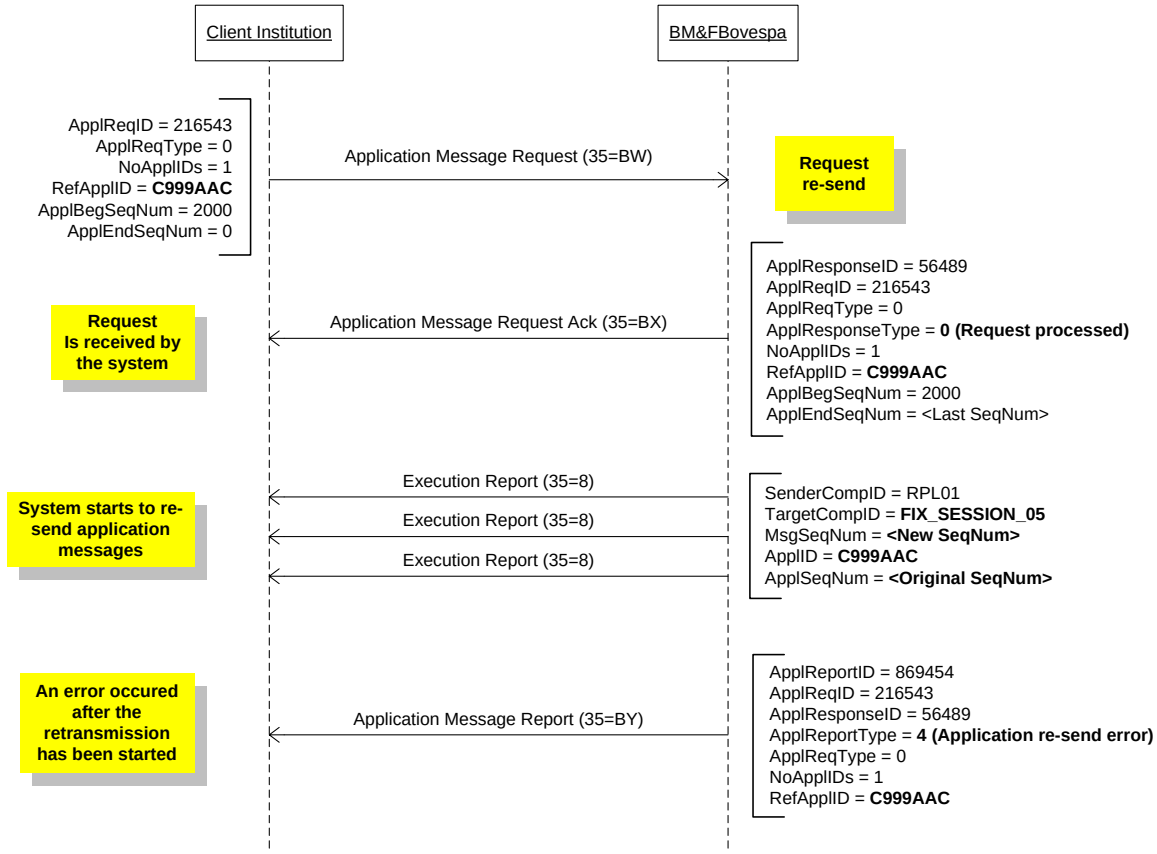

Figure 50 - Error before re-sending

15.8.3.2 Error During Re-sending

At any time during the re-sending, should an error occur, the system may interrupt the retransmission of messages. In this case, an Application Message Report will be sent to the client with tag ApplReportType = 4 and the retransmission is stopped.

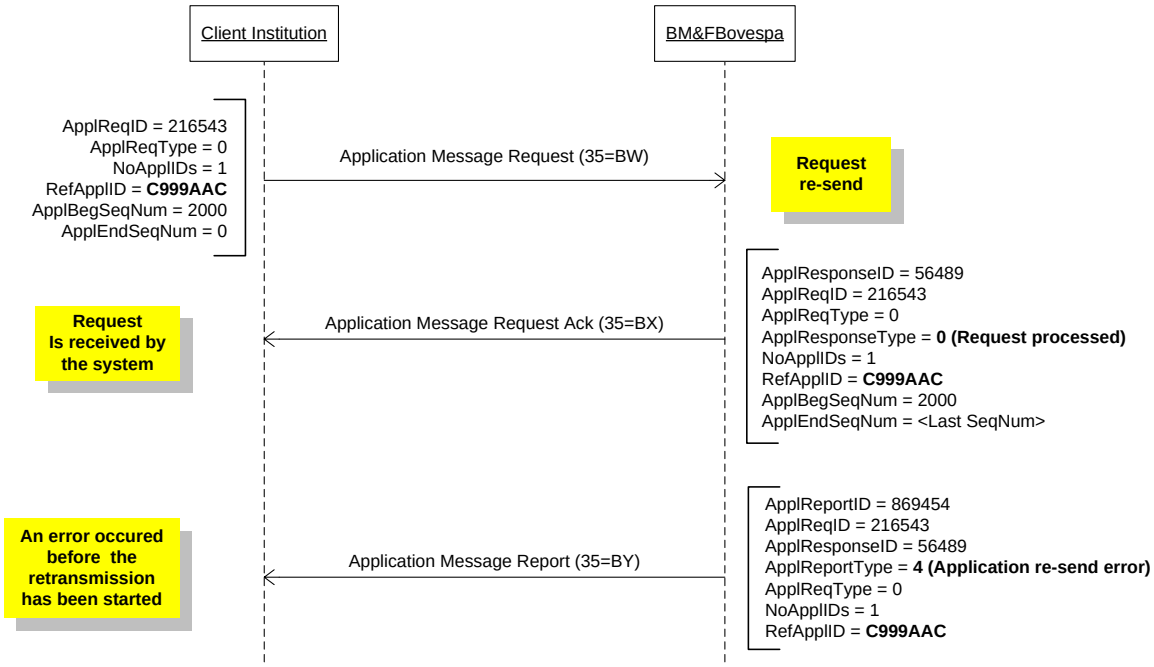

Figure 51 - Error during re-sending

15.9 Market Protections

In the following examples, we assume that instruments XPTO1, XPTO2 and XPTO3 belong to the same group 10. Additionally, consider that the Market Protections have already been triggered for product group 10.

In such scenario, the system will attempt to cancel all resting orders associated with instruments of group 10 and the customer will need to reset the protection before new orders can be accepted for that group again.

15.9.1 Protected Mode

The Protected Mode is triggered for a group of instruments where the protection threshold is reached or, in some cases, exceeded.

15.9.1.1 Automatic Order Cancelation

When the protected mode is triggered, all remaining orders associated with the group are cancelled, except orders related to instruments that are in a state that does not allow order cancelations.

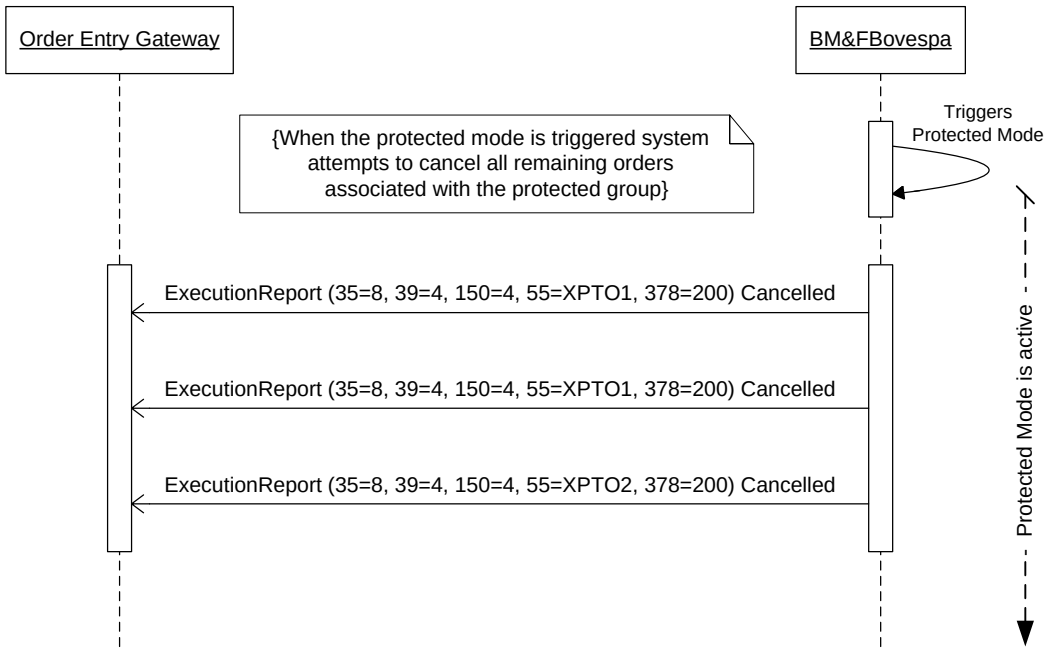

Figure 52 – Remaining Orders are Cancelled when Protection Mode is triggered

The events are described in the table below:

#	Description
1	The Protected Mode is triggered for instruments of group 10
2	System attempts to cancel all remaining resting orders for Group 10

15.9.1.2 Rejection Message

In Protected Mode, the trading platform will prevent the entry of new orders for any instrument associated with the protected group.

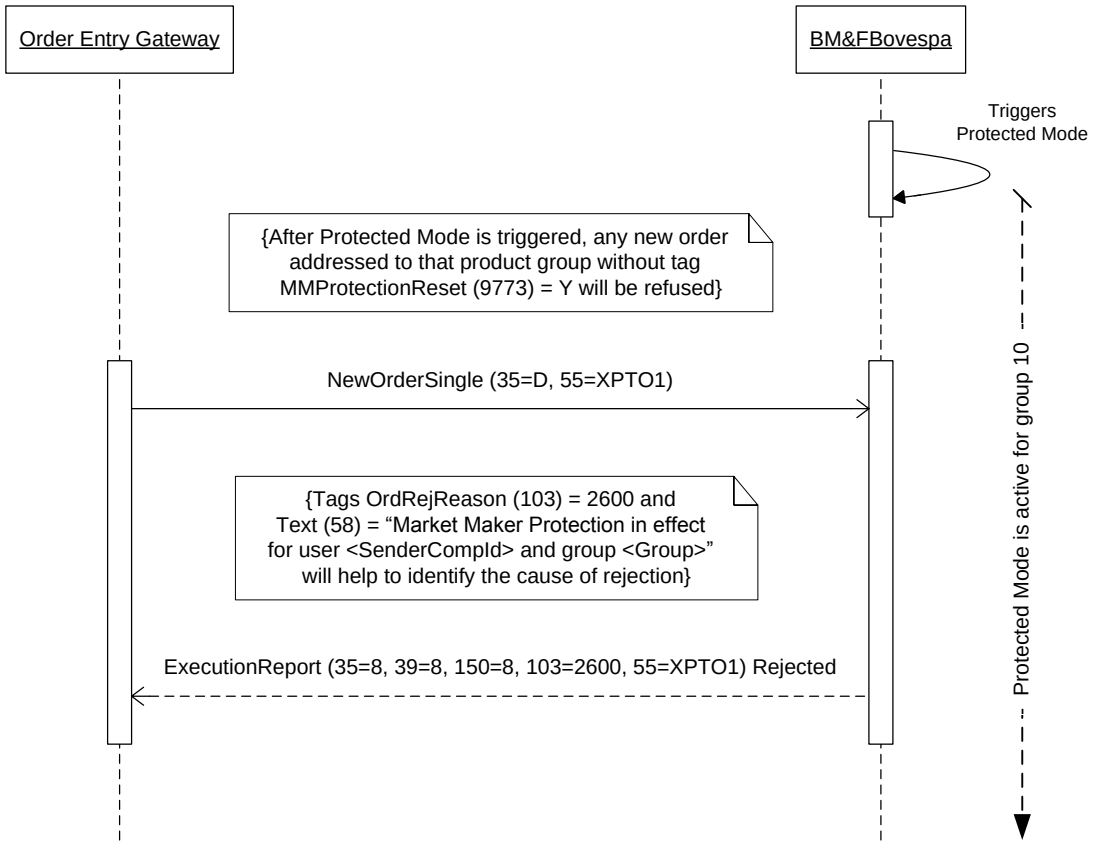

Figure 53 – During Protected Mode New Orders are Rejected

The events are described in the table below:

#	Description
1	Market Protections have already been triggered for product group 10
2	Customer sends a New Order Single (35=D) without tag (9773=Y)
3	Since the group 10 is running in Protected Mode, the new order is rejected

When the customer is ready to start resubmitting orders to that group again, he will need to send a specific indication in the message to reset the monitoring mode.

15.9.2 Resetting Monitoring Mode

In order to reset the protection, the customer must send tag MMProtectionReset (9773) = Y in either a NewOrderSingle (35=D) or OrderCancelReplaceRequest (35=G) message. The following examples depict these two scenarios.

15.9.2.1 Sending MMProtectionReset in NewOrderSingle

In this example, the customer resets the protection by sending a NewOrderSingle (35=D) message with tag MMProtectionReset (9773) = Y.

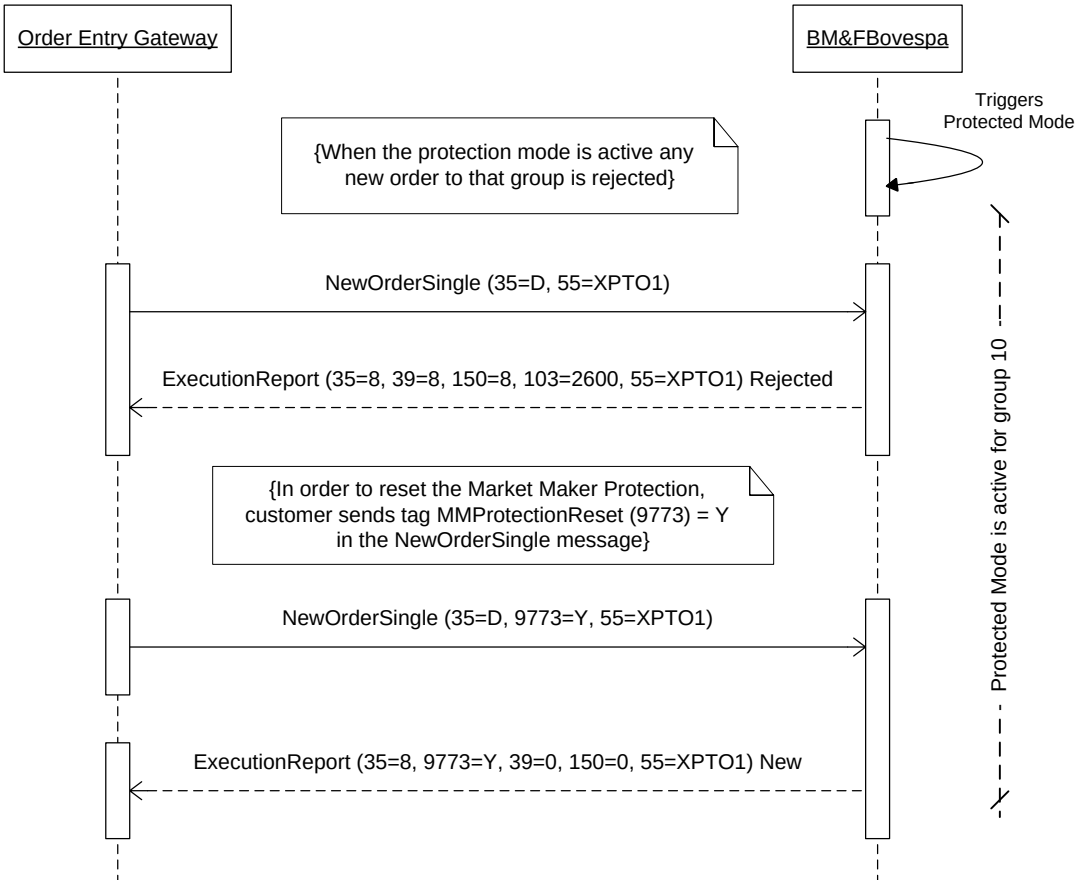

Figure 54 - Resetting Monitoring Mode with New Order Single

The events are described in the table below:

#	Description
1	Market Protections have already been triggered for product group 10
2	Customer sends a New Order Single (35=D) without tag (9773=Y)
3	Since the group 10 is running in Protected Mode, the new order is rejected
4	Customer sends a New Order Single (35=D) with the specific tag (9773=Y) to notify the trading platform to leave the Protection Mode and reset the Monitoring Mode
5	Order is registered
6	New orders for group 10 start being accepted again

15.9.2.2 Sending MMProtectionReset in OrderCancelReplaceRequest

Alternatively, customer may include tag MMProtectionReset (9773) = Y in an OrderCancelReplaceRequest (35=G) message sent to modify an existing order of group 10.

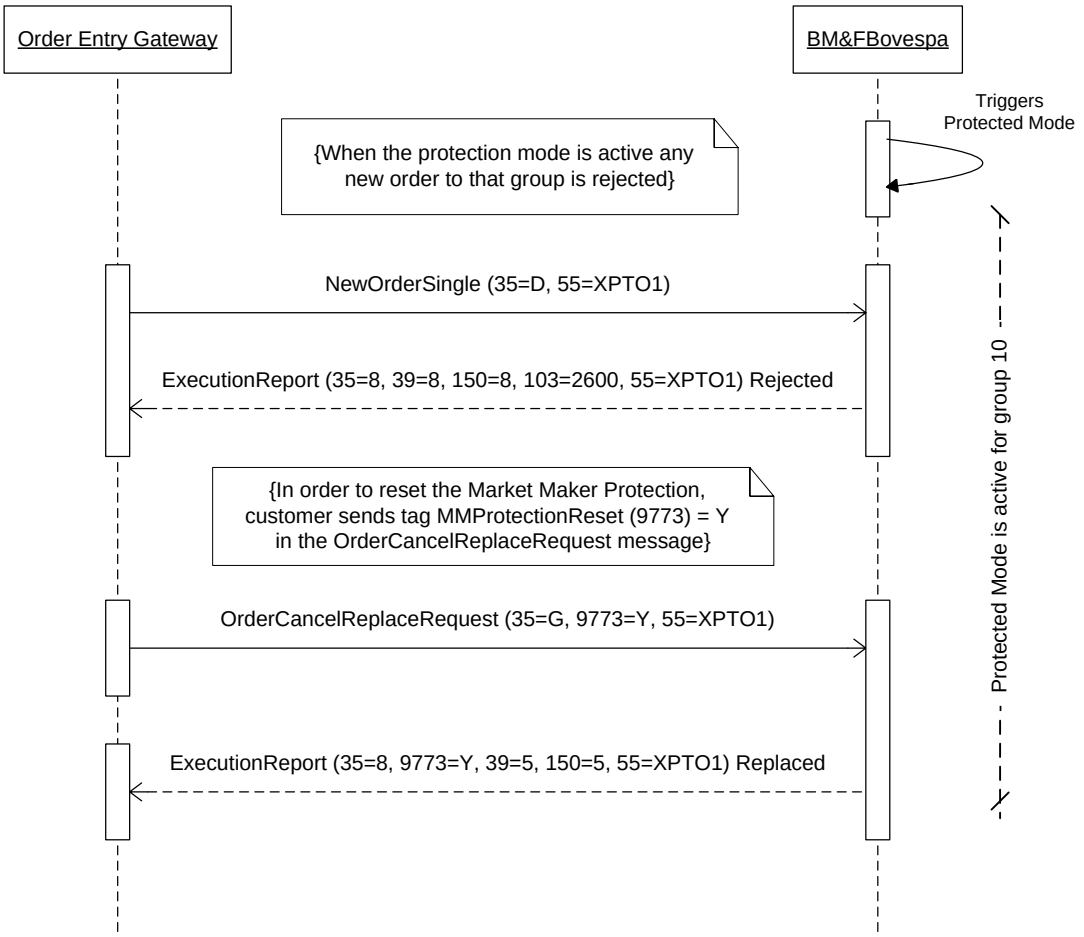

Figure 55 - Resetting Monitoring Mode with Order Cancel Replace Request

The events are described in the table below:

#	Description
1	Market Protections have already been triggered for product group 10
2	Customer sends a New Order Single (35=D) without tag (9773=Y)
3	Since the group 10 is running in Protected Mode, the new order is rejected
4	Customer sends an Order Cancel Replace Request (35=G) with the specific tag (9773=Y) to notify the trading platform to leave the Protection Mode and reset the Monitoring Mode
5	Modification is registered
6	New orders for group 10 start being accepted again

15.9.3 **Order Filled During the Protected Mode**

For this example, consider there is an order of 100 shares for the instrument XPTO3 that is participating in the auction when the Protected Mode is triggered.

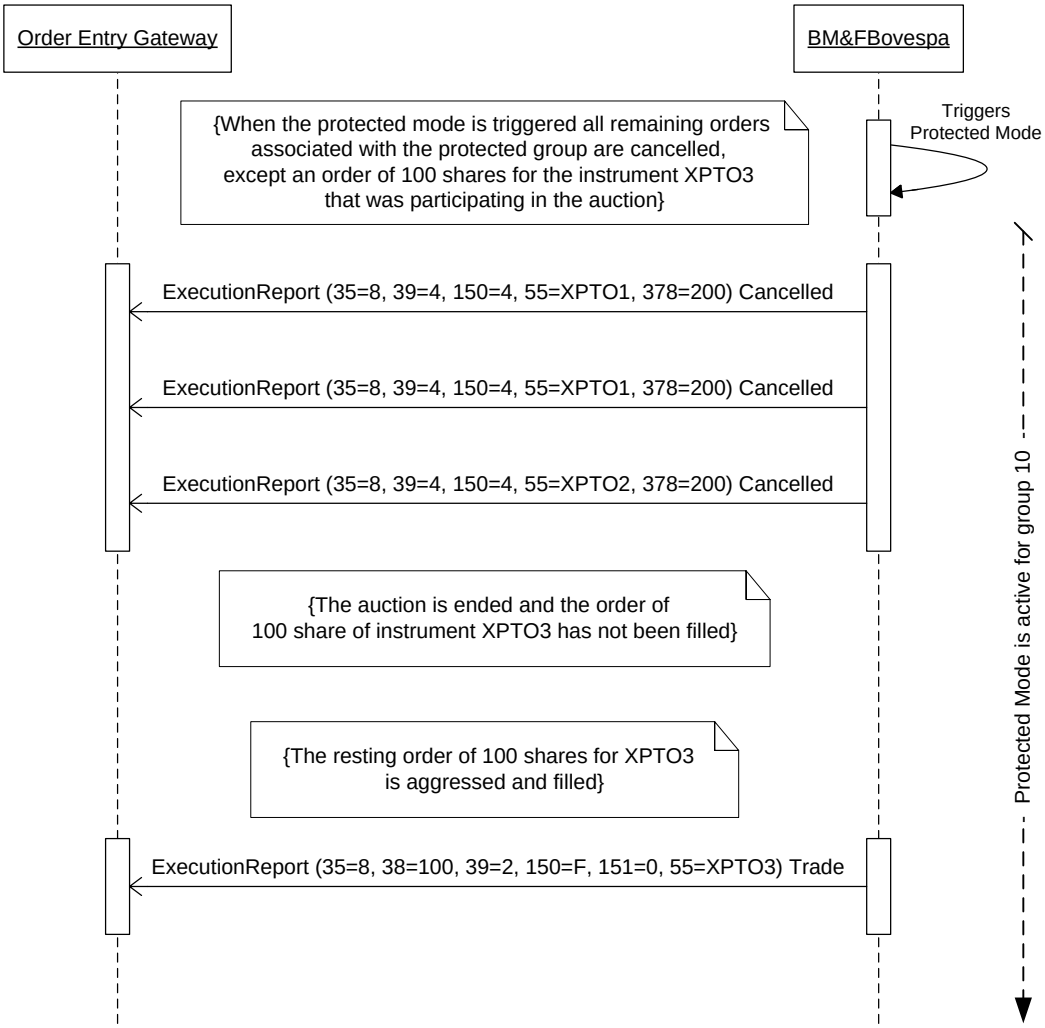

Figure 56 - Order Filled During the Protected Mode

The events are described in the table below:

#	Description
1	The Protected Mode is triggered for instruments of group 10
2	All remaining resting orders of group 10 are cancelled, except an order of 100 shares for the instrument XPTO3 that is participating in the auction
3	The auction for instrument XPTO3 is ended
4	The order of 100 shares for XPTO3 was not filled during the auction
5	Protected Mode continues active for group 10
6	The resting order of 100 shares for XPTO3 is aggressed and filled

15.9.4 **Order Partially Filled during Protected Mode and Remaining Quantity Cancelled**

For this example, consider there is a resting order of 500 shares for instrument XPTO3. The instrument is in *Reserved* state, participating in the auction's theoretical price formation.

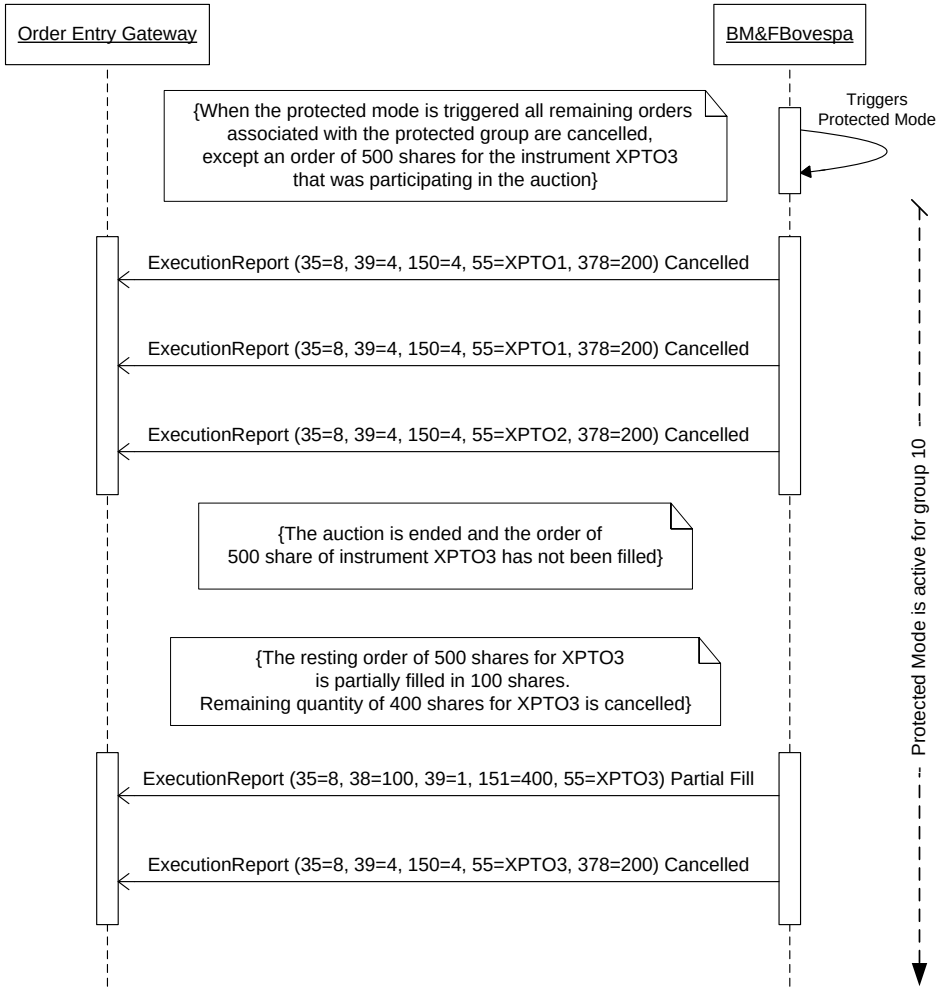

Figure 57 - Order Partially Filled and Remaining Quantity Cancelled

The events are described in the table below:

#	Description
1	The Protected Mode is triggered for instruments of group 10
2	All remaining resting orders of group 10 are cancelled, except an order of 500 shares for the instrument XPTO3 that is participating in the auction
3	The auction for instrument XPTO3 is ended
4	The order of 500 shares for XPTO3 was not filled during the auction
5	Protected Mode continues active for group 10
6	The resting order of 500 shares for XPTO3 is aggressed and partially filled in 100 shares
7	Remaining quantity of 400 shares for XPTO3 is cancelled

15.9.5 **Order Filled and Protection value Exceeded**

This is example depicts a situation where the Protection value is exceeded before the Protected Mode is triggered.

Consider a scenario where a Traded Quantity Protection for Group 10 is set to 1000.

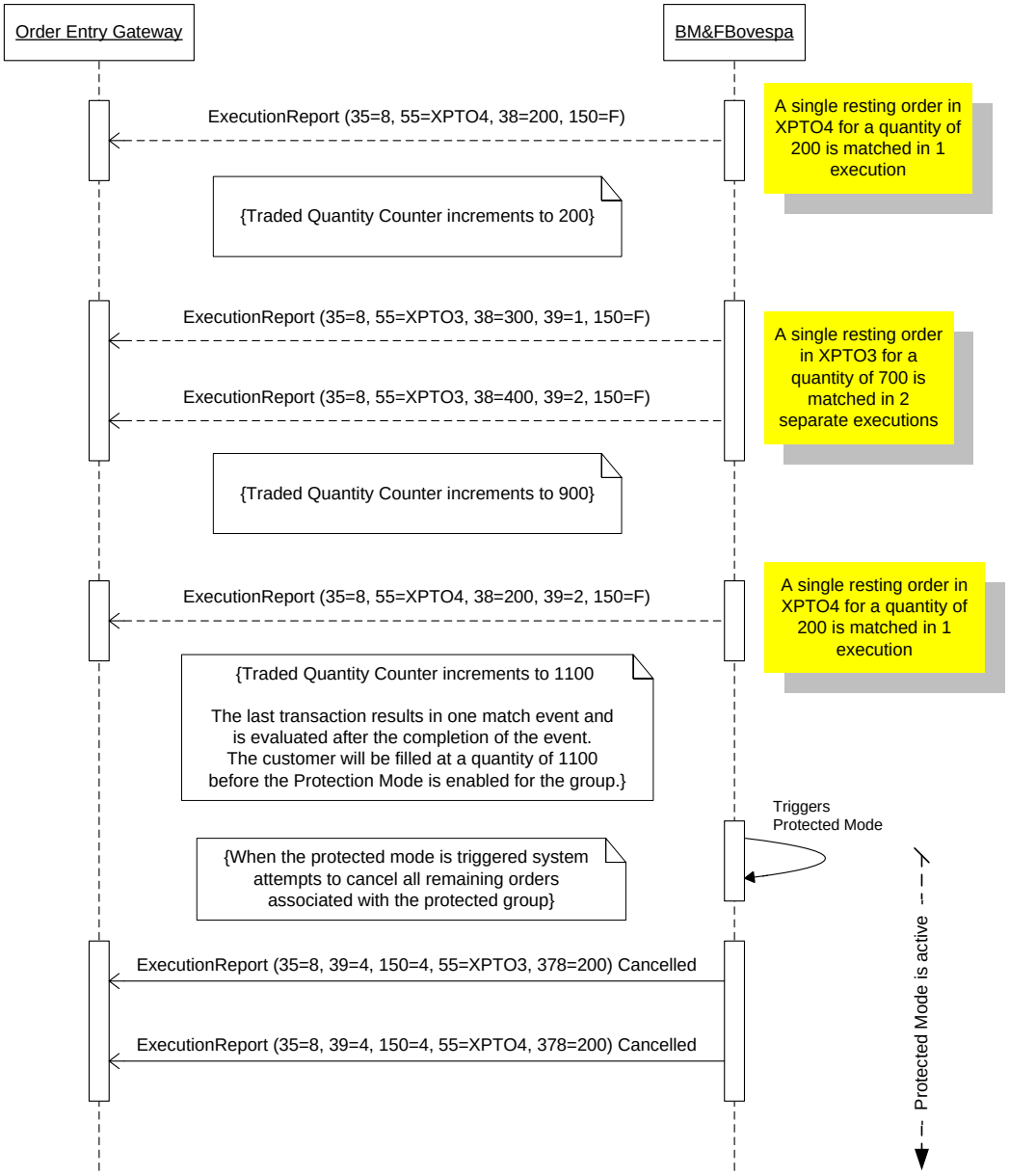

Figure 58 - Order Filled and Protection value Exceeded

The events are described in the table below:

#	Event
1	A single resting order's bid in XPTO4 for a quantity of 200 is matched in 1 execution.
2	A single resting order's ask in XPTO3 for a quantity of 700 is matched in 7 separate executions.
3	A single resting order's ask in XPTO4 for a quantity of 200 is matched in 1 execution.
4	Customer is filled at a quantity of 1100 before the Protection Mode is enabled for the group.
5	Protection mode is enabled for Group 10. All remaining resting orders for Group 10 are cancelled.

15.9.6 **Stop Order Triggered after Auction Not Cancelled at Protection Mode Activation**

For this example, consider there is a stop order of 100 shares for the instrument XPTO3. The instrument XPTO3 is in auction.

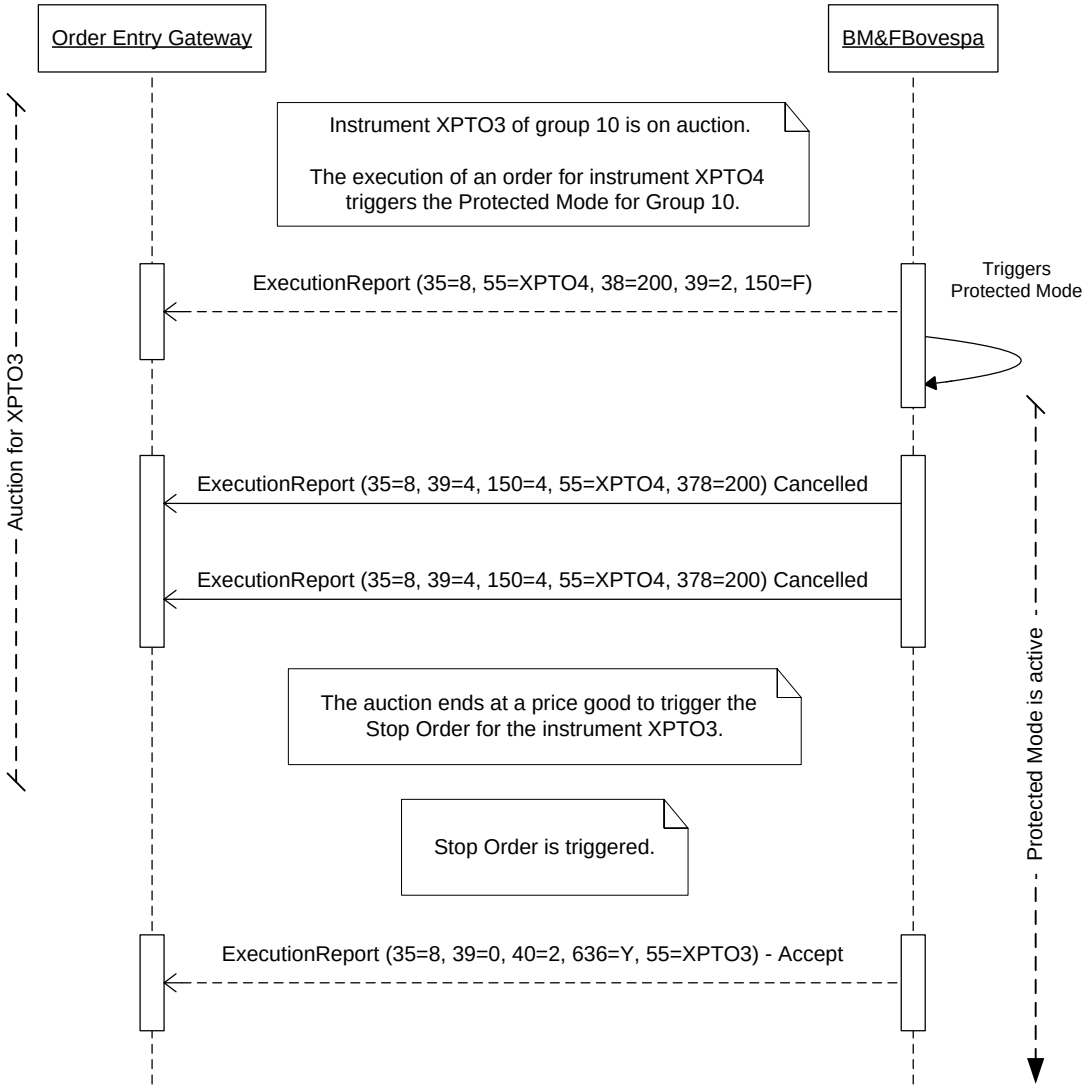

Figure 59 - Stop Order not cancelled at Protection Mode activation

The events are described in the table below:

#	Event
1	The execution of an order for instrument XPTO4 triggers the Protected Mode for Group 10.
2	All remaining resting orders for Group 10 are cancelled except a Stop Order of 100 shares for the instrument XPTO3.
3	The auction ends at a price good to trigger the Stop Order for the instrument XPTO3.
4	The Stop Order is triggered and rests in the book.
5	Protected Mode is still active for Group 10.

Appendix A: **Glossary**

Term	Definition
BM&FBOVESPA	Securities, Commodities and Futures Exchange, based in São Paulo, Brazil. For more information, visit the web site: http://www.bmfbovespa.com.br/en-us/home.aspx?idioma=en-us.
Broker	A broker is an individual or firm who acts as an intermediary between a buyer and seller, usually charging a commission.
Brokerage	Used interchangeably with broker when referring to a firm rather than an individual. Also called brokerage house or brokerage firm.
Correspondent Broker	Identifies a correspondent broker (broker/firm who originates the order to BM&FBOVESPA from a DMA provider or order routing solution).
Counterparty	Party to a trade.
Derivatives	A financial security (such as option or future) whose characteristics and value are derived from the characteristics and the value of another asset.
DMA	Direct Market Access – functionality that allows end-customers, such as hedge funds or investment banks, to directly access the exchange electronically without the need to go over physical broker firm infrastructure.
Entering Firm	Broker who has recorded or reported an execution. This term is particularly useful where the trade is entered into a trading system by a broker who is not a party to the trade, as it allows any inquiries or problem resolution to be directed to the appropriate source.
Entering Trader	Individual usually identified by a trading badge number or initials that actually enters an order to a market (especially in open outcry markets). Usually the Entering Trader is the same as the Executing Trader. However, under some circumstances the Entering Trader will have the trade executed by another trader who is then identified as the Executing Trader.
EntryPoint	BM&FBOVESPA's solution for accessing its electronic trading platform
Executing Firm	Identifies executing / give-up broker.
Executing Trader	Trader or broker id associated with Executing Firm who actually executes the trade.
FIX	Financial Information Exchange Protocol
FIX Gateway	Service that provides connectivity to third-party clients and brokerages using the FIX protocol.
Futures	Contracts covering the sale of financial instruments or physical commodities for future delivery on a commodity exchange.
Give-up firm	Firm to which a trade is given up, i.e. firm which carries the trade.
Holder	The investor holding the right to exercise an option.
Instrument	Financial capital in a readily tradable form.
IP	Internet Protocol
Issuer	An entity that puts a financial asset in the marketplace. The grantor of an options contract who assumes the obligation, if the holder exercises the option, to sell the underlying asset to or buy it from the holder.
Market Data	A collective term for quotes, last sales, volume statistics and other information used by the market to evaluate trading opportunities.
Matching	The process by which two counter-parties that have engaged in a trade compare the settlement details of the trade provided by both. Matching is done to verify all aspects of a trade and ensure that all parties agree on the terms of the transaction.
Order Gateway	The service provided by BM&FBOVESPA which relays FIX messages from a third-party client, usually a vendor, to the SISBEX system.
Order Origination Firm	Firm that originates the order.
Position	Balance resulting from one or more operations with options from the same series, executed on behalf of the same investor, through the same brokerage

Term	Definition
	firm.
Security	A stock, bond or contract that has been authorized for trading on, and by, a registered exchange. Each exchange has different criteria to determine a security's eligibility for listing.
SISBEX	The system used by BM&FBOVESPA to negotiate public bonds.
SSL	Secure Socket Layer
Strike Price	Price at which the holder will be entitled to buy or sell the option's underlying asset.
TCP	Transport Control Protocol
UDS	User-Defined Spread.
Vendor	Institution that sells services to its clients. In the context of this document, a vendor is an institution that sells access to market data feeds and order management interfaces to an Exchange.

Appendix B: **ExecType and OrdStatus transitions**

Scenario	Messages		ExecType	OrdStatus	Remarks
	Incoming	Outgoing			
Order Entry; Accepted	35=D	35=8	New (0)	New (0)	
Order Entry; Rejected	35=D	35=8	Rejected (8)	Rejected (8)	
Modification; Accepted	35=G	35=8	Replace (5)	Replaced (5)	
Cancelation; Accepted	35=F	35=8	Cancelled (4)	Cancelled (4)	
Modification; Rejected (known order)	35=G	35=9	N/A	Standing state	
Modification; Rejected (unknown order)	35=G	35=9	N/A	Rejected (8)	
Cancelation; Rejected (known order)	35=F	35=9	N/A	Standing state	
Cancelation; Rejected (unknown order)	35=F	35=9	N/A	Rejected (8)	
Full Fill	N/A	35=8	Trade (F)	Filled (2)	
Partial Fill	N/A	35=8	Trade (F)	Partially Filled (1)	
Trade Bust (Working order)	N/A	35=8	Trade Cancel (H)	Standing state	
Trade Bust (Non-working order)	N/A	35=8	Trade Cancel (H)	Previous Final State (Z)	
FAK partially filled; Elimination	N/A	35=8	Cancelled (4)	Cancelled (4)	
FOK partially filled; Elimination	N/A	35=8	Cancelled (4)	Cancelled (4)	

Stop Order entry; Ack (no triggering)	35=D	35=8	New (0)	New (0)	Working Indicator (636) = N
Stop order Triggers	N/A	35=8	New (0)	New (0)	Working Indicator (636) = Y
Order with "On Close" attribute is sent; Ack	35=D	35=8	New (0)	New (0)	Working Indicator (636) = N
Order with "On Close" attribute is activated when the closing auction starts	N/A	35=8	New (0)	New (0)	Working Indicator (636) = Y
Iceberg Restatement	N/A	35=8	Restatement (D)	Standing State	
MinQty order entry; Rejected (not enough quantity)	35=D	35=8/35=8	New (0) / Cancelled (4)	New (0)/ Cancelled (4)	
Order Expiration (All TimeInForces except FOK and IOC)	N/A	35=8	Expired (C)	Expired (C)	

Appendix C: **QuoteStatus transitions**

Scenario	Messages		QuoteStatusReportType	QuoteStatus
	Incoming	Outgoing		
Quote Request; Accepted	35=R	35=AI	New (0)	Pending (10)
Quote; Accepted	35=S	35=AI	Accept (1)	Accept (0)
Quote Rejected; Quote Request Found	35=S	35=AI	Reject (3)	Pending (10)
Quote Rejected; Quote Request Not Found	35=S	35= AI	Reject (3)	Quote Not Found (9)
Quote Cancel; Accepted	35=Z	35= AI	Cancel (2)	Cancelled (17)
Quote Cancel Rejected; Quote Request Found	35=Z	35= AI	Reject (3)	Pending (10)
Quote Cancel Rejected; Quote Request Not Found	35=Z	35= AI	Reject (3)	Quote Not Found (9)
Quote Request Reject; Accepted	35=AG	35= AI	Pass(5)	Pass (11)
Quote Request Reject; Rejected; Quote Request Found	35=AG	35= AI	Reject (3)	Pending (10)
Quote Request Reject; Rejected; Quote Request Not Found	35=AG	35= AI	Reject (3)	Quote Not Found (9)
Quote Request Eliminated	N/A	35= AI	Expired (4)	Expired (7)

Appendix D: **Security Strategy types**

Spread Types		
Spread	**Type**	**Description**
3-Way	3W	A 3-Way (3W) option spread is constructed of calls and puts on the same contract and expiry month with three different strike prices. A Call 3-way consists of buying the call for the middle strike price, selling the call for high strike price, and selling the put for the low strike price. A Put 3-way consists of buying the put for middle strike price, selling the put for low strike price, and selling the call for the high strike price. **Spread ratio: (Buy 1: Sell 1: Sell 1)** **3-Way Call Spread** Construction: • Buy 1 Call at strike2exp1 • Sell 1 Call at strike3exp1 • Sell 1 Put at strike1exp1 Example: • Buy 1 ACMEG11C002100 • Sell 1 ACMEG11C002150 • Sell 1 ACMEG11P002050 **3-Way Put Spread** Construction: • Buy 1 Put at strike2exp1 • Sell 1 Put at strike1exp1 • Sell 1 Call at strike3exp1 Example: • Buy 1 ACMEG11P002100 • Sell 1 ACMEG11P002050 • Sell 1 ACMEG11C002150
3-Way: Straddle versus a Call	3C	A 3-way: Straddle versus Call (3C) option spread consists of buying a Straddle and (versus) selling a Call in the same expiry month. The Straddle component consists of buying a Call and buying a Put in the same contract, expiration, and strike price. The opposing (versus) component is to sell a Call for the same contract and expiration but at a different strike price. **Spread ratio: (Buy 1: Buy 1: Sell 1)** Construction: • Buy 1 Call at strike1exp1 • Buy 1 Put at strike1exp1 • Sell 1 Call at strike2exp1 Example: Buy the 3-way: Straddle versus Call • Buy 1 ACMEG11C001900 • Buy 1 ACMEG11P001900 • Sell 1 ACMEG11C002100

3-Way: Straddle versus a Put	3P	A 3-way: Straddle versus Put (3P) option spread consists of buying a Straddle and (versus) selling a Put in the same expiry month. The Straddle component consists of buying a Call and buying a Put in the same contract, expiration, and strike price. The opposing (versus) component is to sell a Put for the same contract and expiration but at a different strike price. **Spread ratio: (Buy 1: Buy 1: Sell 1)** Construction: • Buy 1 Call strike1exp1 • Buy 1 Put at strike1exp1 • Sell 1 Put at strike(?)exp1 Example: Buy the 3-way: Straddle versus Put • Buy 1 ACMEG11C001900 • Buy 1 ACMEG11P001900 • Sell 1 ACMEG11P001700
Box	BX	A Box (BX) option spread consists of buying the call and selling the put at the same lower strike price and buying the put and selling the call at the same higher strike all within the same contract and expiry month. **Spread ratio: (Buy 1: Sell 1: Buy 1: Sell 1)** Construction: • Buy 1 Call at strike1exp1 • Sell 1 Put at strike1exp1 • Buy 1 Put at strike2exp1 • Sell 1 Call at strike2exp1 Example: • Buy 1 ACMEG11C001900 • Sell 1 ACMEG11P001900 • Buy 1 ACMEG11P002100 • Sell 1 ACMEG11C002100

Butterfly	BO	A Butterfly is an options strategy involving three strike prices that are of equal distance apart with all having the same expiration date. **Call Butterfly** Involves buying one call at the lowest strike price, selling two calls at the middle strike price, and buying one call at the highest strike price. **Put Butterfly** Involves buying one put at the highest strike price, selling two puts at the middle strike price, and buying one put at the lowest strike price. Examples: **Call Butterfly** (all Call options) • Leg 1 = Buy 1 ACMEG11C001800 • Leg 2 = Sell 2 ACMEG11C001900 • Leg 3 = Buy 1 ACMEG11C002000 **Put Butterfly** (all Put options) • Leg 1 = Buy 5 ACMEG11P002000 • Leg 2 = Sell 10 ACMEG11P001900 • Leg 3 = Buy 5 ACMEG11P001800 Butterflies are always done in a 1x2x1 ratio. The three Strike prices are always of equal distance apart. All options have the same expiration date.

Christmas Tree (Xmas Tree)	XT	A Xmas Tree (XT) option spread is constructed of all calls (Call Xmas Tree) or all puts (Put Xmas Tree). The Call Xmas Tree consists of buying a call at one strike, selling a call at a higher strike and selling yet another call at a higher strike, all within the same contract and expiration month. The Put Xmas Tree consists of buying a put at a higher strike and selling a put at a lower strike and selling yet another put at a still lower strike, all within the same contract and expiration month. The Xmas Tree requires a specific symmetry in the strikes in that the difference between the strike prices is the same for all legs. **Spread ratio: (Buy 1: Sell 1: Sell 1)** **Call Xmas Tree** Construction: • Buy 1 Call at strike1exp1 • Sell 1 Call at strike2exp1 • Sell 1 Call at strike3exp1 Example: • Buy 1 June 2008 Eurodollar 9800 Call • Sell 1 June 2008 Eurodollar 9850 Call • Sell 1 June 2008 Eurodollar 9900 Call **Put Xmas Tree** Construction: • Buy 1 Put at strike3exp1 • Sell 1 Put at strike2exp1 • Sell 1 Put at strike1exp1 Example: • Buy 1 June 2008 Eurodollar 9900 Put • Sell 1 June 2008 Eurodollar 9850 Put • Sell 1 June 2008 Eurodollar 9800 Put

Conditional Curve	CC	Conditional Curve (CC) option spreads are unique to CME Interest rate products and consists of buying a call (put) at a strike in one instrument group and selling a call (put) at a strike in another instrument group. Additionally, it is possible to have a Conditional Curve spread with a single strike (i.e. same for each leg) or two different strikes, where both strikes are listed. **Call Conditional Curve** Construction: • Buy1callstrikeexp1 instrument1 • Sell1callstrikeexp1 instrument2 Example: • Buy 1 December Eurodollar 9800 Call • Sell 1 December 1-year Mid-Curve 9800 Call **Put Conditional Curve** Construction: • Buy1putstrikeexp1 instrument 1 • Sell1putstrikeexp1 instrument 2 Example: • Buy 1 December Eurodollar 9800 Put • Sell 1 December 1-year Mid-Curve 9800 Put
Condor	CO	A Condor is an options strategy that has four legs (either all Calls or all Puts) and four strikes that are an equal distance apart. **Call Condor** • Involves buying one Call at the lowest strike, selling one Call at the second strike, selling one Call at the third strike, and finally buying one Call at the highest strike. **Put Condor** • Involves buying one Put at the highest strike, selling one Put at the third strike, selling one Put at the second strike, and finally buying one Put at the lowest strike. Example: Buying Call Condor • Leg 1 = Buy 1 ACMEG11C001800 • Leg 2 = Sell 1 ACMEG11C001850 • Leg 3 = Sell 1 ACMEG11C001900 • Leg 4 = Buy 1 ACMEG11C001950 All legs are done in equal quantities. All four Strike prices are an equal distance apart. All options have the same expiration date.

Diagonal	DG	A strategy involving the simultaneous buying and selling of two options of the same type that have different strike prices and different expiration dates. • Two different strikes are used. • Option with the longer expiration date is always bought. • Option with the shorter expiration date is always sold. • Either Strike price (higher or lower) can be bought as long as it has the longer expiration. • Same rules for Calls or Puts. Example: • Leg 1 = Buy 1 ACMEG11C001800 (Buying Feb. 2011 expiration) • Leg 2 = Sell 1 ACMEV10C002100 (Selling Oct. 2010 expiration) Or • Leg 1 = Buy 1 ACMEG11C002100 (Buying Feb. 2011 expiration) • Leg 2 = Sell 1 ACMEV10C002000 (Selling Oct. 2010 expiration) Basically the same trades above but showing the order of the strike prices does not matter as long as they are buying the further out expiration and selling the closer term expiration.
Guts	GT	A Guts (GT) option spread consists of buying a Call at a strike price and buying a Put at a higher strike price in the same expiry. **Spread ratio: (Buy 1: Buy 1)** Construction: • Buy 1 Call at strike1exp1 • Buy1 Put at strike2exp1 Example: Buy the Guts • Buy 1 ACMEG11C002100 • Buy 1 ACMEG11P002200
Horizontal	HO	A horizontal (HO) option spread consists of buying a call (put) at a strike in the far month, and selling a call (put) at the same strike in the near month. **Spread ratio: (Buy 1: Sell 1)** Example • Leg 1 = Buy 1 ACMEG11C001800 • Leg 2 = Sell 1 ACMEV10C001800 Both legs use same quantity and strike price but have different expirations.

Double	DB	A Double (DB) option spread is constructed of all calls (Call Double) or all puts (Put Double). The **Call Double** consists of buying a call at a strike price and buying another call at a higher strike price within the same contract and expiry month. The **Put Double** consists of buying a put at a strike price and buying another put at a lower strike price within the same contract and expiry month. **Spread ratio: (Buy 1: Buy 1)** **Call Double** Construction: • Buy1 Call at strike1exp1 • Buy1 Call at strike2exp1 Example: • Buy 1 ACMEG11C002100 • Buy 1 ACMEG11C002150 **Put Double** Construction: • Buy 1 Put at strike2exp1 • Buy 1 Put at strike1exp1 Example: • Buy 1 ACMEG11P002150 • Buy 1 ACMEG11P002100
Iron Butterfly	IB	An Iron Butterfly contains four legs as is an equivalent strategy to a regular butterfly spread which contains only three legs. Uses three different strikes with both inside legs using the same strike. Construction: • Sell 1 Put at lowest strike • Buy 1 Put at middle strike • Buy 1 Call at middle strike • Sell 1 Call at highest strike Example: Buying Iron Butterfly • Leg 1 = Sell 1 ACMEG11P001800 • Leg 2 = Buy 1 ACMEG11P001850 • Leg 3 = Buy 1 ACMEG11C001850 • Leg 4 = Sell 1 ACMEG11C001900

Iron Condor	IC	An Iron Condor (IC) option spread consists of buying a put spread and buying a call spread at higher strike prices. More specifically this consists of selling a put at one strike price, buying a put at a higher strike price, buying a call at a higher strike price, and selling a call at an even higher strike price, all within the same contract and expiration. **Spread ratio: (Sell 1: Buy 1: Buy 1: Sell 1)** Construction: • Sell 1 Put at strike1exp1 • Buy 1 Put at strike2exp1 • Buy 1 Call at strike3exp1 • Sell 1 Call at strike4exp1 Example: • Sell 1 ACMEG11P001800 • Buy 1 ACMEG11P001850 • Buy 1 ACMEG11C001900 • Sell 1 ACMEG11C001950
Horizontal Straddle	HS	A Horizontal Straddle (HS) option spread consists of buying a straddle at one strike price in the deferred month and selling a straddle at the same or different strike in the near month. More specifically, a Horizontal Straddle (HS) consists of buying a call and buying a put at the same strike price in the deferred month and selling a call and selling a put at the same lower strike price in the near month, all within the same contract and expiry month. **Spread ratio: (Buy 1: Buy 1: Sell 1: Sell 1)** Construction: • Buy 1 Call at strike1exp2 • Buy 1 Put at strike1exp2 • Sell 1 Call at strike1exp1 • Sell 1 Put at strike1exp1 Example: Horizontal Straddle • Buy 1 ACMEG11C002100 • Buy 1 ACMEG11P002100 • Sell 1 ACMEV10C002100 (same or different strike in near month) • Sell 1 ACMEV10P002100 (same or different strike in near month) Or • Buy 1 Sept 2008 Eurodollar 98000 Call • Buy 1 Sept 2008 Eurodollar 98000 Put • Sell 1 June 2008 Eurodollar 98500 Call • Sell 1 June 2008 Eurodollar 98500 Put

Jelly Roll	JR	A Jelly Roll is created by entering into two separate positions simultaneously. One position involves buying a put and selling a call with the same strike price and expiration. The second position involves selling a put and buying a call. The strike prices in the far expiry (second position) can be, but need not be, equal to the strike price in the near expiry. The strike prices of the put and call in the second position are identical but different from the previous position, and the duration of the second position is longer than the previous position. **Jelly Roll (all same strikes, different expirations)** • Sell 1 Call and Buy 1 Put at exp1 • Buy 1 Call and Sell 1 Put at exp2 **Jelly Roll (different strikes, different expirations)** • Sell 1 Call and Buy 1 Put at strike1 at exp1 • Buy 1 Call and Sell 1 Put at strike2 at exp2 Or • Sell 1 Call and Buy 1 Put at strike2 at exp1 • Buy 1 Call and Sell 1 Put at strike1 at exp2
Ratio 1x2	12	A Ratio 1x2 is the purchase of an option(s), call or put, and the selling of a greater number of the same type of options that are out-of-the-money with respect to those purchased. All options involved have the same expiration date. **Call** • Buy 1 Call at strike1 and exp1 • Sell 2 Calls at strike2 and exp1 **Put** • Buy 1 Put at strike2 and exp1 • Sell 2 Puts at strike1 and exp1 Trades can be done in larger quantities if in a 1x2 ratio.
Ratio 1x3	13	A Ratio 1x3 is the purchase of an option(s), call or put, and the selling of a greater number of the same type of options that are out-of-the-money with respect to those purchased. All options involved have the same expiration date. **Call** • Buy 1 Call at strike1 and exp1 • Sell 3 Calls at strike2 and exp1 **Put** • Buy 1 Put at strike2 and exp1 • Sell 3 Puts at strike1 and exp1 Trades can be done in larger quantities if in a 1x3 ratio.

Ratio 2x3	23	A Ratio 2x3 is the purchase of an option(s), call or put, and the selling of a greater number of the same type of options that are out-of-the-money with respect to those purchased. All options involved have the same expiration date. **Call** • Buy 2 Calls at strike1 and exp1 • Sell 3 Calls at strike2 and exp1 **Put** • Buy 2 Puts at strike2 and exp1 • Sell 3 Puts at strike1 and exp1 Trades can be done in larger quantities if in a 2x3 ratio.
Risk Reversal	RR	An option strategy combining the simultaneous purchase of out-of-the-money Calls (Puts) with the sale of out-of-the money Puts (Calls). The options will have the same expiration date and are done in the same quantities. **Call** • Buy 1 Call at strike2 (out-of-money) and exp1 • Sell 1 Put at strike1 (also out-of-money) and exp1
Straddle Strips	SS	Straddle Strips are options strategies which are basically buying
Straddle	ST	A Straddle (ST) option spread consists of buying both a call and put option on the same contract, strike price, and expiration date. **Spread ratio: (Buy 1: Buy 1)** Construction: • Buy 1 Call at strike1 and exp1 • Buy 1 Put at strike1 and exp1 Example: • Buy 1 ACMEG11C001900 • Buy 1 ACMEG11P001900
Strangle	SG	A Strangle (SG) option spread consists of buying a Put at a lower strike price and buying a Call at a higher strike price within the same contract and expiration. **Spread ratio: (Buy 1: Buy1)** Construction: • Buy 1 Put at strike1exp1 • Buy 1 Call at strike2exp1 Example: • Buy 1 ACMEG11P001900 • Buy 1 ACMEG11C001950

Strip	SR	A Strip (SR) option spread is constructed of all Calls (Call Strip) or all Puts (Put Strip). The Call Strip consists of buying calls within the same contract and strike price for each of four consecutive quarterly expiry months, resulting in a total of four (4) Calls purchased. The Put Strip consists of buying puts within the same contract and strike price for each of four consecutive quarterly expiry months, resulting in a total of four (4) Puts purchased. The Strip requires a specific symmetry in the expiry months in that the time difference between the expiry months is the same for all legs. **Spread ratio: (Buy 1: Buy 1: Buy 1: Buy 1)** **Call Strip** Construction: • Buy 1 Call at strike1exp1 • Buy 1 Call at strike1exp2 • Buy 1 Call at strike1exp3 • Buy 1 Call at strike1exp4 Example: Call • Buy 1 June 2008 Eurodollar 9800 Call • Buy 1 Sept 2008 Eurodollar 9800 Call • Buy 1 Dec 2008 Eurodollar 9800 Call • Buy 1 March 2009 Eurodollar 9800 Call **Put Strip** Construction: • Buy 1 Put at strike1exp1 • Buy 1 Put at strike1exp2 • Buy 1 Put at strike1exp3 • Buy 1 Put at strike1exp4 Example: Put • Buy 1 June 2008 Eurodollar 9800 Put • Buy 1 Sept 2008 Eurodollar 9800 Put • Buy 1 Dec 2008 Eurodollar 9800 Put • Buy 1 March 2009 Eurodollar 9800 Put

Vertical	VT	A Vertical (VT) option spread is made up of all calls or all puts and consists of buying a call at a strike price and selling a call at a higher strike price or buying a put at a strike price and selling a put at a lower strike price within the same contract and expiration date. **Spread ratio: (Buy 1: Sell 1)** **Call Vertical** Construction: • Buy 1 Call at strike1exp1 • Sell 1 Call at strike2exp1 Example: Call Spread • Buy 1 ACMEG11C001900 • Sell 1 ACMEG11C001950 **Put Vertical** Construction: • Buy 1 Put at strike2exp1 • Sell 1 Put at strike1exp1 Example: Put Spread • Buy 1 ACMEG11P001950 • Sell 1 ACMEG11C001900
Cash / Cash	VV	A strategy involving the simultaneous buying and selling of two cash instruments (Equity vs. Equity, Equity vs. ETF, or ETF vs ETF) Example: • Leg 1 = Buy 1 ACME3 (Buying ACME common stock) • Leg 2 = Sell 1 ACME4 (Selling ACME preferred stock) Or • Leg 1 = Buy 1 ACME3 (Buying ACME common stock) • Leg 2 = Sell 1 BOVA11 (Selling IBovespa ETF)
Cash / Option	VO	A strategy involving buying or selling any number (greater than 0) of Cash instruments (Equity or ETF) and selling or buying any number (greater than 0) of equity options. No more than 40 total instruments allowed in a single strategy.
Single Stock Future / Option	FO	A strategy involving buying or selling any number (greater than 0) of Single Stock Futures (Equity or ETF) and selling or buying any number (greater than 0) of equity options. No more than 40 total instruments allowed in a single strategy.
Single Stock Futures / Single Stock Future	FF	A strategy involving the simultaneous buying and selling of two Single Stock Future instruments Example: • Leg 1 = Buy 1 ACMEEFX (Buying ACME May Single Stock Future) • Leg 2 = Sell 1 ACMEFFX (Selling ACME June Single Stock Future)

Cash / Single Stock Future	FV	A strategy involving the simultaneous buying a Cash Equity and selling a Single Stock Future instrument Example: • Leg 1 = Buy 1 ACME3 (Buying ACME common stock) • Leg 2 = Sell 1 ACMEEFX (Selling ACME May Single Stock Future)
Generic	GN	Any UDS that does not fit any of the above listed strategy types will be assigned the type of Generic.

Appendix E: **Order Characteristics - Allowed Combinations (Equities)**

Not all order characteristics can be combined on all circumstances. The following table summarizes how the various order parameters can be combined:

Order Characteristics		Types					Quantities		Validities							Misc.
		Limited	Market to Limit	Stop Limit	Market	Stop	Disclosed	Minimum	Day	FAK	FOK	GTC	GTD	MOC	MOA	Auction
Types	Limited						✓	✓	✓	✓	✓	✓	✓			✓
	Market to Limit							✓	✓	✓	✓	✓	✓			
	Stop Limit						✓		✓	✓	✓	✓	✓			
	Market								✓	✓	✓	✓	✓	✓	✓	✓
	Stop								✓	✓	✓	✓	✓			
Quantities	Disclosed	✓		✓					✓			✓	✓			
	Minimum	✓	✓						✓	✓	✓	✓	✓			
Validities	Day	✓	✓	✓	✓	✓	✓	✓								✓
	FAK	✓	✓	✓	✓	✓		✓								✓
	FOK	✓	✓	✓	✓	✓		✓								
	GTC	✓	✓	✓	✓	✓	✓	✓								✓
	GTD	✓	✓	✓	✓	✓	✓	✓								✓
	MOC				✓											✓
	MOA				✓											✓
Misc.	Auction	✓			✓				✓	✓		✓	✓	✓	✓	

Appendix F: **Order Characteristics - Allowed Combinations (Derivatives)**

Not all order characteristics can be combined on all circumstances. The following table summarizes how the various order parameters can be combined:

Order Characteristics		Types					Quantities		Validities							Misc.
		Limited	Market to Limit	Stop Limit	Market	Stop	Disclosed	Minimum	Day	FAK	FOK	GTC	GTD	MOC	MOA	Auction
Types	Limited						✓	✓	✓	✓	✓	✓	✓			✓
	Market to Limit						✓	✓	✓	✓	✓	✓	✓			
	Stop Limit								✓			✓	✓			
	Market						✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Stop								✓			✓	✓			
Quantities	Disclosed	✓	✓		✓				✓	✓	✓	✓	✓			
	Minimum	✓	✓		✓				✓	✓						
Validities	Day	✓	✓	✓	✓	✓	✓	✓								✓
	FAK	✓	✓		✓		✓	✓								✓
	FOK	✓	✓		✓		✓									
	GTC	✓	✓	✓	✓	✓	✓									✓
	GTD	✓	✓	✓	✓	✓	✓									✓
	MOC				✓											✓
	MOA				✓											✓
Misc.	Auction	✓			✓				✓	✓		✓	✓	✓	✓	

Appendix G: **Order Characteristics - Allowed Modifications**

Matrix of changes among order types

The possibilities of change are shown in the table below:

FROM \ TO	Limited	Market Order	Market with protection	Stop with Protection	Stop
Limited		✓	✓	✓	✓
Market Order					
Market with Protection					
Stop with Protection	✓	✓	✓		✓
Stop	✓	✓	✓	✓	

Matrix of changes among order qualifiers

The possibilities of change are shown in the table below:

FROM \ TO	Day	GTC	GTD	FAK	FOK	MOC	MOA[8]
Day		✓	✓	✓	✓		✓
GTC	✓		✓	✓	✓		✓
GTD	✓	✓		✓	✓		✓
FAK							
FOK							
MOC	✓	✓	✓	✓	✓		✓
MOA	✓	✓	✓	✓	✓	✓	

[8] An order may have its validity modified to MOA only if the instrument is in auction or if the market is in Pre-Open or Final Closing Call.